

DOCUMENT RESUME

ED 211 073

IR 009 893

AUTHOR Newton, David J.; And Others
TITLE How to Run a Computer and Keep Up with It. A Co-Ser Project.
INSTITUTION Cortland-Madison Board of Cooperative Educational Services, Cortland, N.Y.
PUB DATE 81
NOTE 124p.; Best copy available.
EDRS PRICE MF01/PC05 Plus Postage.
DESCRIPTORS *Computer Literacy; Computers; *Computer Science Education; Elementary Secondary Education; History; Instructional Materials; *Microcomputers; *Programming; *Programming Languages
IDENTIFIERS *BASIC Programming Language

ABSTRACT

These curriculum materials are designed for teachers of students in grades six through nine to use to teach general computer literacy and rudimentary programming in BASIC. Sections on history, what microcomputers can do, and computer anatomy have been written so they can be copied and placed in the hands of the students in the intended audience, together with the programming lessons, which include objectives, activities, and suggested exercises. The 11 lessons review the beginning commands and steps in using a computer and show how to write programs to print a statement, calculate using arithmetic, use variables, note differences in the order of operations, input numbers, input and output alphanumeric variables, use REM, TAB, and GOTO statements, decisions processes, loops, and counters and/or adders. A list of 22 BASIC commands, the names and descriptions of 46 prepared programs, and the answers to the lesson exercises follow the lessons. Also included are ten quizzes and a final quiz, 60 enrichment exercises, instructions for a computer quiz maker program, directions for writing a program for multiple choice quizzes, a glossary of 54 computer terms, and instructions for the care and handling of diskettes. These materials were prepared for use with four types of microcomputers: APPLE, HP-2000, PET, and TRS-80. (CHC)

 * Reproductions supplied by EDRS are the best that can be made *
 * from the original document. *

ED211073

U.S. DEPARTMENT OF EDUCATION
NATIONAL INSTITUTE OF EDUCATION
EDUCATIONAL RESOURCES INFORMATION
CENTER (ERIC)

X This document has been reproduced as
received from the person or organization
originating it.

□ Minor changes have been made to improve
reproduction quality.

• Points of view or opinions stated in this docu-
ment do not necessarily represent official NIE
position or policy.

HOW TO **RUN** A COMPUTER*



•AND KEEP UP WITH IT!

A CO-SEP PROJECT

CORTLAND-MADISON BOCES

SUPPLEMENT

BEST AVAILABLE
COPY

Written by participants from:

Cincinnati Central School: Tom Rausa
Cortland Enlarged City Schools: Douglas W. Green
DeRuyter Central School: Debbie Skeele
Homer Central Schools: Ted Larison
Marathon Central Schools: Norma Parker
McGraw Central Schools: John R. Troy

Consultant/leader: David J. Newton

"PERMISSION TO REPRODUCE THIS
MATERIAL HAS BEEN GRANTED BY

Michael Bay-Borelli

TO THE EDUCATIONAL RESOURCES
INFORMATION CENTER (ERIC)."

IR009893

PREFACE

The purpose of this project was to prepare curriculum materials that could be used to teach general computer literacy and rudimentary programming in BASIC. The intended audience for these materials is teachers of students in grades six through nine. We feel, however, that these materials could be beneficial to anyone from a gifted student in any elementary grade to an adult who wishes to learn more about the subject matter covered here. The lessons in programming in BASIC, along with the sections on history, what microcomputers can do, and computer anatomy have been written so that they can be copied and placed in the hands of the students in our intended audience. It remains for the teacher to decide if the students can handle the material as it is presented. In some cases, the class should only attempt to read this material after it has been presented during class discussions. In other situations, it might be presented to a student on any age to be studied on an independent basis. As such, therefore, it also represents a self learning package for a teacher who is not experienced in this area. For anyone with a little bit of experience, the sample quiz programs can be used to prepare quizzes on any subject. However, they are meant to be learning experiences. We caution anyone against using them to generate grades. To help you while you try to construct your own quizzes, please run and list the sample quizzes on the disks that are included with this package.

We have also included sample programs that deal with low resolution graphics and sound. These are not topics that have been included in the BASIC programming lessons but we know that they will be of interest to the more capable students in your classes. Don't feel like you have to understand these programs before you place them in the hands of students who are interested in them. In this field, unlike others, the students routinely surpass their teachers in their ability to compose programs. The wise teacher uses capable students to help with their own learning. In any event, it would be a big mistake to wait until you feel confident with the microcomputer before you use it in the class room. You might find that you have waited while the rest of the world has passed you by. In order to get started, try the disk that contains the TUTOR program. This will help anyone learn to use the microcomputer. It does not require any knowledge of programming. This also points out the fact that anyone can use the micro without ever learning anything about writing their own programs. This package has been prepared by a group of experienced teachers who also have experience in programming, writing, and using many different computer systems. If you have any comments or suggestions, please feel free to contact any of the authors to let them know what you think.

The material provided in this project was produced for the four types of computers found in schools in the Cortland-Madison B.O.C.E.S. district. They are:

APPLE---Trademark of APPLE Computer Company.

HP-2000---Trademark of Hewlett-Packard.

PET---Trademark of Commodore Business Machines, Inc.

TRS-80---Trademark of Radio Shack, Tandy Corporation.

TABLE OF CONTENTS

Preface

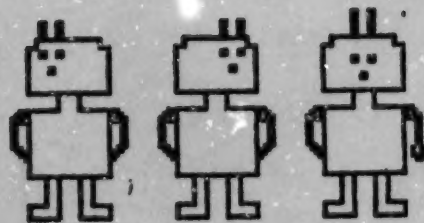
1 From Bones To Electronic Fingers	1
2 I'm More Than Just A Piece Of Silicon	7
3 Running With The Program	12
4 They Only Do What They're Told	17
4-1 A Look At Someone Else's Computer	18
4-2 Good Morning Computer	21
4-3 A Very Expensive Calculator	22
4-4 LET There Be X	24
4-5 Which Comes First ?	27
4-6 Talking Back	32
4-7 Plenty Of \$\$\$\$'s But No Money!!	35
4-8 Running Up A TAB To REMember!	38
4-9 Decisions, Decisions	44
4-10 Getting Looped	48
4-11 Meet The Electronic Fingers	54
5 The Amazing Microcomputer	58
6 Other Goodies !!!	63
6-1 List Of Basic Commands	64
6-2 Prepared Software	65
6-3 Answers To Exercises	68
6-4 Prepared Quizzes From Chapter 4	88
6-5 Enrichment Exercises	103
6-6 Computer Quiz Makers	108
7 BUZZZZZZZZZZZZZZWORDS	112

FROM BONES

TO

ELECTRONIC

FINGERS I I



FROM BONES TO ELECTRONIC FINGERS

Man has always searched for aids in performing mathematics. Perhaps the first and most readily available aid employed by our ancestors was the fingers. Our base-ten number system certainly owes its popularity to the number of fingers available for counting. Since we generally do not consider our hands as "devices", they probably should not be called computers.

Perhaps the earliest device for storing numbers was the abacus. This dates back to about 3000 B.C. The abacus does no calculations itself. It helps a person record and remember numbers as calculations are done.

In the seventeenth century, several major achievements in computing devices were accomplished. In 1617, Lord Napier invented a mechanical device to aid in the solution of certain problems. It was called Napier's Bones. In 1642, Blaise Pascal designed the first mechanical adding machine. In the late seventeenth century, Gottfried Leibnitz improved Pascal's adding machine. He produced a device that could also multiply and divide. This device was the forerunner of today's desk calculator.

The first major tool for storing information was developed at the beginning of the nineteenth century. Joseph Marie Jacquard, a weaver of fabrics, perfected a method of controlling looms using

small cards made of stiff paper. Each card was punched with holes that controlled the action of hooks on the loom. Once the card had been punched, it would always make the loom produce a repeatable pattern in the fabric. The cards were strung together in a sequence so the loom would produce the same pattern over and over again. This was one of the first instances in which a machine operated automatically.

More than eighty years later, the United States Bureau of the Census was getting ready to take the 1890 census. A young worker named Herman Hollerith devised a method for recording and counting the millions of census forms. He used a system similar to Jacquard's loom cards. Hollerith's cards contained holes representing possible responses to census questions. These cards were entered into a machine that counted the holes. With this system, the Bureau completed the 1890 census in two and a half years, one-third the time required for the 1880 count.

The abacus, Jacquard's cards and Hollerith's cards and codes are all memory aids. They help people to record information accurately and to recall what has been recorded. Machines that extend the human ability to process information were also developed during this time period. Pascal and Leibnitz devices fall in this category.

Mathematicians and others could now use machines for all four basic arithmetic operations: addition, subtraction, multiplication, and division. Even though these machines were able to solve some

problems, certain operations were still too difficult for them. Charles Babbage, a British mathematician, noted that the lists of corrections in books of tables were often longer than the tables themselves. In the 1820's, he began plans for a general purpose machine that could calculate a table of any length. His design combined, for the first time, the calculating capabilities of Leibnitz's machine, Jacquard's punched cards, and other storage capabilities. Babbage called it the Analytical Engine. Babbage never finished his work. The failure of Babbage's engine was due to the lack of existing technology of his era. The discovery and application of electricity opened new avenues for computer development.

A machine similar to the Analytical Engine was first built more than a century later, in 1939. This was the Automatic Sequence Controlled Calculator, also called the Mark I computer. Howard Aiken and his co-workers at Harvard University developed the Mark I.

The next major advance after the Mark I was the development of a computing machine based on vacuum tubes. Previous machines had been primarily mechanical, not electrical devices. In 1946 two engineers at the University of Pennsylvania named J. Eckert and J. Mauchly built the first digital computer that was controlled by vacuum tubes. It was called ENIAC (Electronic Numerical Integrator And Calculator). One major disadvantage of ENIAC was its size. The machine was as large as many houses and weighed 30 tons. Surprisingly, this giant had less memory and was slower than some of today's hand-held

calculators.

ENIAC prompted many interesting developments. In 1947 John von Neumann developed the idea of keeping instructions for the computer, inside the computer's memory. Von Neumann also suggested the use of binary numbers rather than decimals for the construction and control of computers.

The basic design of the computer has not changed much since the early 1950's. It still uses electronic signals to do the calculations, and the binary system is universal. In fact, the only real changes have been the speed with which data can be processed and the size of the basic components. This is possible because of two major advances: the transistor and the integrated circuit.

Computers which used the vacuum tube are called First Generation Computers. The Second Generation Computers were those using transistors. These computers could perform work ten times faster than First Generation Computers.

The Third Generation Computers came out in 1965. These computers can do more than a million calculations a second. Third Generation Computers are controlled by tiny integrated circuits and are smaller and more dependable than Second Generation Computers. An integrated circuit is a large number of transistors on a small chip.

Fourth Generation Computers are now produced. The integrated

PAGE 8

circuits have been concentrated into a tiny microprocessor chip no larger than your finger. There are presently chips which will hold 64,000 bits.

So great have been the advances, that today, within the size of a portable typewriter, we can do more computing than the 30 ton ENIAC. How far can we go?

I'M MORE THAN JUST A PIECE OF SILICON

The basic design of a microcomputer system contains the following components:

1. An input device
2. An output device
3. A CPU (Central Processing Unit)
4. Storage

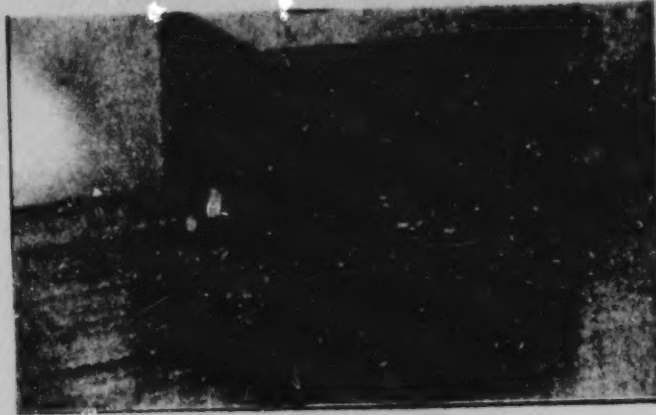
The INPUT DEVICE is the keyboard. The keyboard resembles a typewriter. There are a few keys that are not the same as you would find on a typewriter. In the diagram below two of these keys, ESC and CTRL, are on the left. Their functions are strictly for computer commands. We will explore their purposes later.



Apple Keyboard

The OUTPUT DEVICE is either a T.V. monitor or a printer.

A Monitor



The monitor may be color or black and white. you may also hear the monitor referred to as a CRT (Cathode Ray Tube).



A Printer

There are many types of printers offering a variety of printing styles. They vary in price according to their speed and printing capabilities.

The CPU or central processing unit in the microcomputer is the microprocessor chip. A chip is a small piece of silicon upon which an integrated circuit is built. The purpose of the CPU is to interpret and carry out instructions. In the picture below, the CPU chip is the large rectangular chip in the upper center portion of the board, above the words apple computer inc.

A microcomputer system requires memory or DATA STORAGE to perform any type of function. To store information two types of memory chips are used.

One chip is called the RAM chip. RAM stands for Random Access Memory. These chips store data what you input. When the computer is turned off, the data is erased. The RAM chips are located inside the white square.

The other memory chip is the ROM chip. ROM stands for Read only Memory. These chips keep information indefinitely and you do not lose the information when the power is lost.

ROM chips are used to store many computer directions and commands. The information appearing on the screen after turning on a computer is stored on a ROM chip. The ROM chips are the larger chips directly above the RAM chips and below the CPU.

The most common SECONDARY STORAGE components are cassette tape recorders or disk drive units.



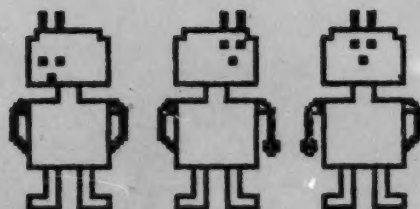
Although cassette tape recorders are less expensive than disk drive units, they are quite slow in saving and retrieving data. A disk system does have the advantage of immediate access to large bodies of data.



INSIDE A DISK DRIVE

RUNNING

WITH THE PROGRAM

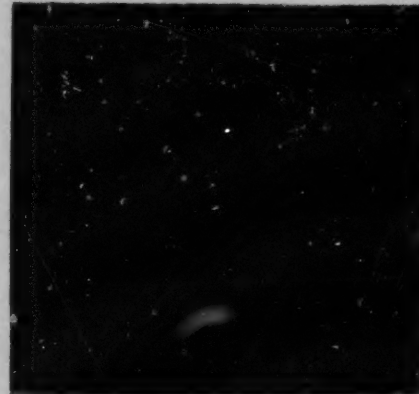


RUNNING WITH THE PROGRAM

You will run a simple program for an Apple computer. If you are using a different type of computer, refer to the List of Basic Commands. Commands for all computers are not the same.

Locate your disk drive and open the door. Insert the disk (carefully) into the slot. If you don't it may bend or center improperly. Never touch the recording surface of the disk!

Close the door.



Turn on the computer. The disk drive will start to run and you will see the red light. The computer is now reading instructions off of the disk. These instructions help the computer find and save programs on the disk. This process is called "Booting". Do not try to remove or insert a disk while the red light is on!



You will now see a
picture on the screen
and hear music.



Next you will see
the catalog. The
catalog is the listing
of all the programs on
this disk.

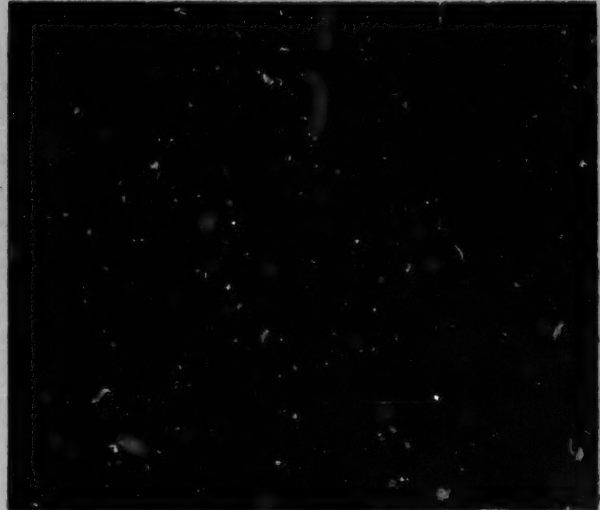
Load the TUTOR program. To do this, type LOAD TUTOR, and hit
return key.

We will now demonstrate how to run the program. Type RUN, and
hit return key. Follow directions on the screen.

Let's clear the screen. Type HOME and push return. The screen is now cleared and the flashing cursor is now in the upper left part of the screen.

Now we will see how the program was made. To do this, we will list the program. Type LIST, and press return. You will see lines numbered from 10 to 130.

Now we will show how to delete a line from the program. Notice line 100. This creates the flashing effect. Line 120 stops the flashing effect.



Run the program again so you remember how it looked. Clear the screen, then list the program.

If we do not want the flashing effect, we can delete lines 100 and 120. This is done by typing the number 100 and pressing the return key. The line numbered 100 is now deleted from the computer (it still remains on the disk). Repeat the operation for line 120.

List your program and notice that lines 100 and 120 are now deleted.

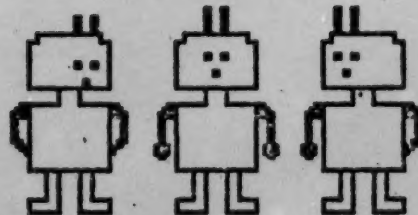
Run the program again. Notice that the last response is no longer flashing.

We will now save the changed program. Save this program under the name CHANGED. To do this type SAVE CHANGED.

I 'M MORE THAN

JUST

A PIECE OF SILICON!!



Catalog the disk. This means to list the names of all of the programs on the disk. This is done by typing CATALOG and pushing return. You will notice that the catalog now has added the program named CHANGED.

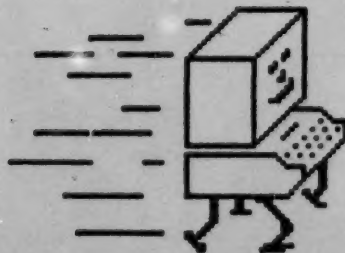
Load the program CHANGED. Type LOAD CHANGED and hit return. list the program. Notice that it is still there.

We will now demonstrate how to delete a program from the computer. This is done by typing the word NEW. Now type NEW. Clear the screen. Now list the program. Notice that the program has been deleted from the computer (but it still remains on the disk). Verify this by cataloging again.

Now we will delete the program named CHANGED from the disk. This is done by typing DELETE CHANGED. Catalog again and note that the program CHANGED has been deleted.

THANK YOU FOR GOING THROUGH THIS LESSON WITH US.

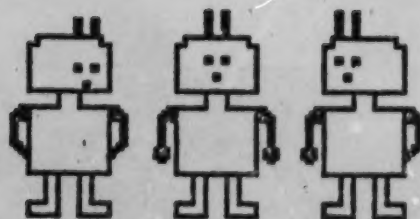
HAPPY COMPUTERING !!



THEY ONLY DO WHAT

WHAT

THEY'RE TOLD !!



4-1 A LOOK AT SOMEONE ELSE'S

OBJECTIVE: The purpose of this section is to review the beginning commands and steps in using a computer. AFTER EACH COMMAND, THE RETURN OR ENTER KEY MUST BE PRESSED.

- 
1. To execute a program, type the word RUN.
 2. To list the lines of a program, type the word LIST.
 3. To stop a list on the screen:
 - a. APPLE--hold CTRL down and hit S.
 - b. TRS-80--hit the BREAK key.
 - c. HP 2000--hit the NO SCROLL key.
 - d. PET--hit the RUN-STOP key.
 4. To continue the printing of a list on the screen:
 - a. APPLE--type hold CTRL down and hit S
 - b. TRS-80--type CONT.
 - c. HP 2000--hit the NO SCROLL key a second time.
 - d. PET--hit the RUN-STOP key a second time.
 5. To list a line of a program, type LIST then the line number.
 6. To delete a line from the program, type the line number.

7. To clear the screen:

- a. APPLE--type HOME.
- b. TRS-80--type CLE or hit the CLEAR key.
- c. HP 2000--not possible
- d. PET--hold the SHIFT key down and hit the CLEAR-HOME key.

8. To erase the current program from the memory:

- a. APPLE, TRS-80, or PET -type NEW.
- b. HP 2000--type SCRATCH.

9. To save a program:

a. on a disk:

- 1. APPLE--type SAVE NAME.
- 2. HP 2000--type SAVE-NAME. NOTE: NAME may be any combination of characters, not exceeding 8 characters.

b. on a cassette:

- 1. APPLE--type SAVE.
- 2. TRS-80--type CSAVE "A". NOTE: Letters A-Z may be used.
- 3. PET--type SAVE "NAME".

10. To find out what programs are stored on a disk, type CATALOG.

11. To load a program:

a. from a disk:

1. APPLE--type LOAD NAME.
2. HP 2000--type GET-NAME.

b. from a cassette:

1. APPLE--type LOAD.
2. TRS-80--type CLOAD "A".
3. PET--type LOAD "NAME".

12. To erase a program:

a. from a disk:

1. APPLE--type DELETE NAME.
2. HP 2000--type KILL-NAME.

b. from a cassette:

1. TRS-80 and PET-- record over.

NOW FOR THE FUN TO BEGIN!!!



4-2 GOOD MORNING COMPUTER

OBJECTIVE: Upon completion of this section you should be able to write a program that would print a desired statement.

A program is a series of instructions telling the computer what to do. The instructions are given to the computer line by line. Since the computer follows the instructions in numerical order, each line of the instructions is numbered. (Generally by multiples of 10.) The word used to tell the computer to write something on the screen is PRINT. The word, number, or sentence that you want to print must be in quotes and placed after the word PRINT. It is proper to end a program with an END statement.

The following program will have the computer print GOOD MORNING on the screen.

```
10 PRINT "GOOD MORNING"
20 END
```

 In summary, we note that anything enclosed by quotes in a PRINT statement is printed exactly as is.

SUGGESTED EXERCISES

1. Write a program that prints the message "THIS IS EASY."
2. Write a program that prints your name and address.
3. Write a program that prints a message of your choice.

4-3 A VERY EXPENSIVE CALCULATOR

OBJECTIVE: Upon completion of this section, you should be able to write a program that would handle a desired calculation using a (+) for addition, (-) for subtraction, (*) for multiplication, (/) for division, and (^) to raise a number to a power.

We note in this section that the computer can be used as a calculator. The following example is a program that prints the sum of 8 and 4, the difference of 8 and 4, the product of 8 and 4, the quotient of 8 and 4, and 2 raised to the 3rd power. You will recall that 2 to the 3rd power is equal to $2*2*2$.

```
10 PRINT 8+4
20 PRINT 8-4
30 PRINT 8*4
40 PRINT 8/4
50 PRINT 2^3
60 END
```



Your output should look like this:

```
12
4
32
2
8
```

Please note the difference in using quotes after PRINT.

```
10 PRINT "10 + 5"
```

```
20 PRINT 10+5
```

```
30 END
```

In line 10, the computer prints 10 + 5.

In line 20, the computer prints 15.

SUGGESTED EXERCISES

1. Write a program that will add, subtract, multiply, and divide 13 and 3.
2. Write a program that will raise 8 to the second (2nd) power.
3. Write a program that will use (+), (-), (*), (/), and (^) at least once.

4-4 LET THERE BE X

OBJECTIVE: Upon completion of this section you should be able to write a program using variables to represent numbers, sums, differences, products, or quotients.

A variable is a letter that stands for a number. The letter X is often used as a variable. The value of X might be 2 or it might be 27. It could be anything! It's value can vary. This is why it is called a variable.

The instruction to the computer that assigns a number to a variable is a LET statement. The following program assigns the value of $10 + 5$ to the variable X, then it prints the value of X.

```
10 LET X=10 + 5
20 PRINT X
30 END
```

Commas and semicolons are used to separate variables when there are more than one variable to be printed.

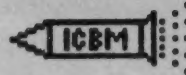
In the next program, notice the difference in using commas as opposed to using semicolons in the PRINT statement.

```
10 LET X=10
20 LET Y=15
30 LET Z=X+Y
```

```
40 PRINT X,Y,Z
```

```
50 PRINT X;Y;Z
```

```
90 END
```



Using a comma between the variables in line 40 will print the results in three columns approximately 15 spaces apart.

```
10      15      25
```

Using a semicolon will not provide for any spacing at all! The results of line 50 will run X,Y, and Z together as follows: 101525

Now you should be ready to put some personal touches in your own programs. For example:

```
10 LET G=8
```

```
20 LET J=3
```

```
30 PRINT "G DIVIDED BY J IS ";G/J
```

```
40 END
```

```
10 LET Q=25
```

```
20 LET P=35
```

```
30 PRINT "THE SUM OF ";Q;" AND ";P;" IS ";Q+P
```

```
40 END
```

SUGGESTED EXERCISES

1. Write a program that assigns 28 and 36 to variables and then adds them.

2. Write a program that assigns 985 and 937 to variables and then multiplies them.
3. Write a program that uses the LET, PRINT, and END statements. Use a semicolon and a comma in your PRINT statement(s).
4. Write a program that will average five (5) numbers of your choice.

4-5 WHICH COMES FIRST ?

OBJECTIVE: Upon completion of this section you should be able to write a program noting the differences in the order of operations.

The four basic mathematical operations are addition, subtraction, multiplication, and division. When the computer reads an expression containing more than one operation, it does not necessarily compute the operations in order from left to right! If there are no parentheses the order is as follows:

1. Multiplication and division in order from left to right.
2. Addition and subtraction in order from left to right.

For example, the expressions $5*6+2$ and $2+5*6$ both equal 32 since multiplication is performed before addition.

The expression $48/8-2*2$ is computed as follows:

$$48/8-2*2=$$

$$6-2*2=$$

$$6-4=$$

$$2$$

What do you think $100-64/8$ is equal to?

If you said 92, YOU ARE RIGHT! We can note here that since division

is computed before the subtraction, we will have:

$$100 - 84/8 =$$



$$100 - 8 =$$

$$92$$

An exponent in an expression will be computed before the four basic operations. (Recall 2 to the 3rd power is typed as 2^3 on the computer.)

$$90 + 2^3 =$$

$$90 + 8 =$$

$$98$$

What do you think $40 - 8/2 + 2 * 1^3$ is equal to?

If you said 38, YOU'VE GOT THE HANG OF IT!



$$40 - 8/2 + 2 * 1^3 =$$

$$40 - 8/2 + 2 * 1 =$$

$$40 - 4 + 2 * 1 =$$

$$40 - 4 + 2 =$$

$$38 + 2 =$$

$$38$$

What do you think $25 * 4 - 3^2 * 6/3 + 4$ is equal to?

If you said 86, YOU'RE RIGHT!



$$25 * 4 - 3^2 * 6/3 + 4 =$$

$$25 * 4 - 9 * 6/3 + 4 =$$

$$100 - 9 * 6/3 + 4 =$$

100-34/3+4=

100-18+4=

82+4=

86

If there are parentheses in the expression, you must perform the operations within the parentheses first.

$4+5*3=16$ but $(4+5)*3=27$.

What do you think $(1+2)*4^2/(8-(2-1))$ is equal to?

If you said 8, YOU'VE GOT IT!



$(1+2)*4^2/(8-(2-1))$

$3*4^2/(8-(2-1))$

$3*4^2/(8-1)$

$3*4^2/8$

$3*16/8$

$48/8$

6

Let's see how this works in a program. Type in:

10 PRINT "3+2^2= ";3+2^2

20 PRINT "(3+2)^2= ";(3+2)^2

30 END

Do you see why you get 7 from line 10 and 25 from line 20?

In line 10:

$3+2^2=$

3+4=

7

In line 20:

(3+2)^2=

5^2=

25

Try this program and see if you can tell what results you get from lines 10-40.

10 PRINT 3+2*4^2

20 PRINT (3+2)*4^2

30 PRINT 3+(2*4)^2

40 PRINT ((3+2)*4)^2

50 END

Your results will be: 35 in line 10, 80 in line 20, 67 in line 30, and 400 in line 40. If you understand this program, you are ready for the suggested exercises.

REMEMBER-This is the order in which the operations are done:

1ST--Anything in parentheses

2ND--Exponents

3RD--Multiplication and Division

4TH--Addition and Subtraction



These operations are all done from left to right.

SUGGESTED EXERCISES

1. Write a program to compute $8*12/3-2*5$. (ans. 22)
2. Write a program to compute $8*12/(3-2)*5$. (ans. 480)
3. Write a program, putting in parentheses where necessary, so that $5*24-2/1+1$ will equal 55.
4. Write a program, putting in parentheses where necessary, so that $96/24/2$ will equal 8. Write a second program so that $96/24/2$ will equal 2.
5. Write a program that evaluates $10-2^3$ and $(10-2)^3$.
6. Write a program showing that subtraction is not associative.
7. Write a program using (+), (-), (*), or (/) and four fours to get the values 0,1,2,3,4,5,6,7,8,9. In some cases, there is more than one way to get the value. For example, one way to get 8 is:

```
10 PRINT ((4*4)-4)-4
```

```
20 END
```


4-8 TALKING BACK

OBJECTIVE: Upon completion of this section you should be able to write a program that allows for the input of a number.

Frequently it is convenient to allow a program to ask for a number. The word **INPUT** tells the computer to wait until something has been typed in. A variable must follow the word **INPUT**.

For example, the following program will square any number that is typed in.

```
10 PRINT "WHAT IS THE NUMBER THAT YOU WANT SQUARED?"  
20 INPUT N  
30 LET S=N*N  
40 PRINT S;" IS THE SQUARE OF ";N  
50 END
```

The computer will first print:

WHAT IS THE NUMBER THAT YOU WANT SQUARED?

Then a question mark will appear on the next line indicating that you are to enter a number. Upon receiving the number the computer will finish the program, computing the square.

It is possible to request more than one number at a time with one **INPUT** statement by using commas.

This program will average any three numbers.

```

10 PRINT "WHAT ARE YOUR THREE NUMBERS?"
20 INPUT X,Y,Z
30 LET A=(X+Y+Z)/3
35 PRINT X,Y,Z
40 PRINT "THE AVERAGE OF YOUR THREE NUMBERS IS ";A
50 END

```

As the computer executes the program, it will first print:

WHAT ARE YOUR THREE NUMBERS?

Then one question mark will be printed on the next line showing that you should now type in your first number. After you enter the first number, two question marks will appear on the screen for the second number to be entered. (Likewise for the third number.) The computer will then finish the program.



SUGGESTED EXERCISES

1. Write a program that will calculate the perimeter of a square. The program will ask for the length of a side and output the perimeter.
2. Bill works and gets \$3.50 an hour. Write a program to tell how much he will get paid for any amount of hours worked. The program will ask for the number of hours Bill worked to be entered.
3. Write a program that will calculate the area of a rectangle. The program will ask for the length and width of the rectangle and will output the area.

4. Mr. Brown always gives 8 tests for each marking period. Write a program that will help him find averages easily.

3. A. Write a program to convert Celsius and Fahrenheit. The program will ask for the degrees in Celsius and tell what it equals in Fahrenheit. ($F = C * 9/5 + 32$)

B. Write a program to convert Fahrenheit to Celsius. The program will ask for the degrees in Fahrenheit and tell what it equals in Celsius. ($C = (F - 32) * 5/9$)

6. Write a program that will calculate the miles per gallon for a car. The program will ask for the number of miles and the number of gallons used.

7. Write a program that will calculate the MPG for a car. This time ask for two odometer readings and the number of gallons.

8. Write a program that will tell you the discount and the price you would pay for an item at a percent of discount. Example: If a \$60 item were sold at a 15% discount, the discount would be \$9 ($60 * .15 = 9$) and the cost would be \$51 ($60 - 9 = 51$).

4-7 PLENTY OF \$\$\$\$'S BUT NO MONEY!!

OBJECTIVE: Upon completion of this section, you should be able to write programs allowing for the input and output of alphanumeric variables.

Variables up to now have been for numbers only. However we can assign words, letters, or characters to a special type of variable called an alphanumeric variable. When you want to input a letter or a word into a program you must use an alphanumeric variable. The symbols used for alphanumeric variables have a letter followed by a \$. Some examples are: A\$, M\$, or Y\$.

When assigning a word, letter, or character to an alphanumeric variable, they must be in quotes as seen in the following example:

```
10 LET Y$="COMPUTER WIZARD"
20 PRINT "I AM A ",Y$
30 END
```

In this program, the words COMPUTER WIZARD are assigned to Y\$. The computer output will look like this:

I AM A COMPUTER WIZARD

Try It! 

The next program allows for the input of a word, letter, or character.


```
10 PRINT "WHAT IS YOUR NAME?"
20 INPUT N$
30 PRINT "HELLO ";N$
40 END
```

You may want to try this several times using different words, characters, and letters.

In the next example, we have a program that inputs a word and then inputs a number.

```
10 PRINT "WHAT IS YOUR NAME?"
20 INPUT N$
30 PRINT "HOW OLD ARE YOU?"
40 INPUT N
50 PRINT "IN TEN YEARS ";N$;" WILL BE ";N+10;" YEARS OLD."
60 END
```

Please note the difference between N\$ and N in this program! They are not the same! N\$ is the alphanumeric variable. N is a variable that can only be assigned a number. A word like Joe or Bob cannot be assigned to a variable such as N.

The value of using alphanumeric variables can be seen in the following program. The program will print a short grade report.

```
10 PRINT "WHAT IS STUDENT'S NAME?"
20 INPUT N$
30 PRINT "WHAT SUBJECT?"
40 INPUT S$
```

```

                PAGE 37
50 PRINT "WHAT WAS THE GRADE?"
60 INPUT G
70 HOME
80 PRINT "DEAR ";N$;","
90 PRINT
100 PRINT " IN THE PAST TEN WEEKS YOUR GRADE FOR "
110 PRINT S$;" WAS ";G;". "
120 PRINT
130 PRINT " YOUR LOVING TEACHER."
140 END

```

Notice that two alphanumeric variables (N\$ and S\$) and the variable G were used in this program. Remember that HOME will clear the screen so that the letter can begin at the top of the screen and the PRINT statement in lines 90 and 120 skip a space.

A program like this can save retyping the same thing over and over for several students. It is a type of form letter.

If N\$="BILL", S\$="ENGLISH", AND G=91, the computer will print the following:

```

DEAR BILL,

IN THE PAST TEN WEEKS YOUR GRADE FOR

ENGLISH WAS 91.

YOUR LOVING TEACHER

```

SUGGESTED EXERCISES

1. Write a program using an alphanumeric variable that will print a simple message.
2. Write a program requesting a name to be input. The output should be a statement including the name that was requested.
3. Write a letter of explanation that tells why your homework is not being handed in on time. BE CREATIVE! Variables should include your teacher, the date and the reason. 
4. Suppose you are working for the No- Value Hardware Store. Write a program that will output a letter to local contractors indicating that a long-awaited hardware shipment has just arrived. Variables should include the name of the contractor, the date, description of the shipment, and cost.
5. You are going to host a party. Write a program that will output an invitation for your party. The date, the time, the place, and the occasion should be printed in the invitation. Variables should include the guest and the guest's address.

4-8 RUNNING UP A TAB TO REMEMBER!

OBJECTIVE: Upon completion of this section you should be able to write a program using REM and TAB statements.

It is often convenient to insert remarks into a program indicating what the program is doing. This can be done with a REM statement. The REM statement will have no effect on a program but it will appear on the listing of the program. For example:

```

10 REM***A PROGRAM TO FIND THE AREA OF A CIRCLE***
20 PRINT "TYPE IN THE RADIUS (IN METERS)"
30 INPUT R
40 LET A=3.14*R*R
50 REM***THE VALUE OF PI BEING USED IS 3.14***
60 PRINT "THE AREA IS ";A;" SQ. METERS"
70 END

```

The computer will print the following if the radius is 8 meters:

TYPE IN THE RADIUS (IN METERS)

?

8.

THE AREA IS 200.96 SQ. METERS

We can note that lines 10 and 40 will not be printed, but will be in the list.

Now It's Your Turn!



Suppose you wanted the character * printed 10 spaces in from the left of the screen. One way to do this is to use a PRINT statement with nine spaces between the left-hand quote and the *.

```
10 PRINT "          *"
20 END
```

A much shorter way is to use a TAB in a PRINT statement as follows:

```
10 PRINT TAB(10); "*"
20 END
```

The TAB is always used in a PRINT statement. You have to tell the computer two main things:

WHERE TO PRINT...WHAT

```
10 PRINT TAB(20); "%"
20 END
```

The above program will print % in the 20th space from the left---Try it and see!



The next example illustrates how we can use two tabs in a PRINT statement.

```
10 LET R=7
20 LET B=8
30 PRINT TAB(10);R+B;TAB(30);R*B
40 END
```

The computer will print:

13

58

(A)

(A)

10th space

30th space

You can also use the TAB to make a design.

```

10 REM THIS PROGRAM WILL PRINT A RECTANGLE
20 PRINT TAB(5);"*****"
30 PRINT TAB(5);" ";TAB(14);" "
40 PRINT TAB(5);" ";TAB(14);" "
50 PRINT TAB(5);" ";TAB(14);" "
60 PRINT TAB(5);" ";TAB(14);" "
70 PRINT TAB(5);" ";TAB(14);" "
80 PRINT TAB(5);"*****"
90 END

```

Give It A Try!



The following program illustrates how TAB can simplify a form letter.

```

10 PRINT "WHAT IS YOUR NAME?"
20 INPUT N$
30 HOME
40 REM'''RECALL THAT HOME CLEARS THE SCREEN'''
50 PRINT "DEAR ";N$;" ,"
60 PRINT TAB(5);"WE REGRET TO INFORM YOU THAT YOU DID
NOT WIN A PRIZE IN OUR NONREADER'S DIGEST SWEEPSTAKES.
HOWEVER, IF YOU ACT NOW YOU CAN TAKE ADVANTAGE OF OUR

```

LOW SUBSCRIPTION RATES. PLEASE SEND YOUR ORDER IN TODAY."

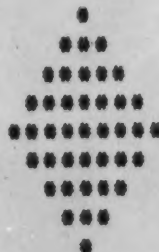
```
70 PRINT TAB(20);"THANK YOU"
80 PRINT TAB(20);"CLYDE GOETTER"
90 PRINT TAB(20);"SUBSCRIPTION MANAGER"
100 PRINT TAB(20);"NONREADER'S DIGEST INC."
110 END
```

Be sure to run this program!

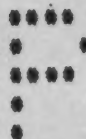


SUGGESTED EXERCISES

1. Write a program that will print your name starting in the 17th space from the left. (Use the word TAB.)
2. Write a program that inputs two numbers. The program will output the first number 7 spaces from the left, the second number 17 spaces from the left, and their product 27 spaces from the left.
3. Write a program using the TAB statement that will print this design:



4. Write a program, using the TAB statement, that will print your initials as a design on the screen. Ex:



5. Write a program, using the TAB statement, that will print a design on the screen.
6. Write a program for a form letter using the TAB statement to announce the birth of your first child. The program should have the child's name and date of birth as alphanumeric variables.
7. You are an employee of the Party University (P.U.) ticket office. You have to send out letters to 3000 people who have purchased season footbrawl tickets. The letters should include the seat location, the date of the first game and the amount of money owed on the tickets. Write a program to do this. Include TAB and REM statements. (Be sure to include an appropriate heading!)

4-8 DECISIONS, DECISIONS, ...

OBJECTIVES: (1.) Upon completion of this section you should be able to write a program using GOTO statements. (Also be able to break out of a program that is running in a continuous loop.)

(2.) You should be able to write a program involving a decision process. The program will use an IF...THEN statement.

A GOTO statement directs the computer to go to the designated line.

```
10 PRINT "HELLO"  
20 PRINT TAB(20); "GOODBYE"  
30 GOTO 10  
40 END
```

The above example will continuously print HELLO and GOODBYE.

 BREAKING OUT OF A PROGRAM

To break out of the program on an APPLE computer, hit the CTRL and RESET keys.

To break out of the program on a TRS-80 computer, hit the BREAK key.

To break out of the program on an HP 2000 computer, hit the BREAK key.

To break out of the program on a PET computer, hit the RUN-STOP key.

Suppose we want to print out all the squares of the natural numbers from 1 to 20. We cannot do this easily with the statements we have learned so far. To do this, an IF...THEN statement may be used. The statement begins with an IF followed by some condition. A condition compares two things. The things being compared are separated by one of these comparison symbols.

EQUAL...=

NOT EQUAL...(<)

LESS THAN...(<

LESS THAN OR EQUAL TO...(<=

GREATER THAN...>)

GREATER THAN OR EQUAL TO...>=)

The condition is followed by the word THEN and a line number. If the condition is true, the computer proceeds to the line number after the word THEN. If the condition is false, the computer proceeds directly to the next line.

The following program will print the numbers 1 through 20, and their squares.

```
10 LET N=1
20 PRINT N, N*N
30 IF N=20 THEN 60
40 LET N=N+1
45 REM---THE VALUE OF N IS INCREASED BY 1---
50 GOTO 20
60 END
```



- Line 10 starts with $N=1$.
- Line 20 will print N and its square.
- Line 30 is your check. If $N=20$ then you want the program to end so the computer will skip to line 60 and end. If N is less than 20, the computer will go to the next line of the program, in this case 40.
- Line 40 will add one to N to make it the next higher natural number. Thus 2 becomes 3, 17 becomes 18, and so on in this line.
- Line 50 will send the computer back to line 20 to print the next higher value of N and its square.

IF...THEN statements are very powerful. They force the computer to make a decision. In line 30, the computer has to decide to go on to line 40 or skip to line 60. When N is any other value than 20, the computer goes to line 40. When N is 20, it goes directly to line 60.

The next program will request two numbers, and then it will request their product.

```
5 REM+++THIS PROGRAM IS A MULTIPLICATION QUIZ+++  
10 PRINT "PICK ANY NUMBER"  
20 INPUT A  
30 PRINT "PICK ANOTHER NUMBER"  
40 INPUT B
```

PAGE 47

```
50 PRINT "WHAT IS ";A;" TIMES ";B
60 INPUT C
65 REM+++STATEMENT 70 COMPARES C TO A*B+++
70 IF C=A*B THEN 100
80 PRINT "WRONG! TRY AGAIN"
90 GOTO 50
100 PRINT "CORRECT!"
105 REM+++CHECK TO SEE IF OPERATOR+++
108 REM+++WANTS TO RUN PROGRAM AGAIN+++
110 PRINT "WANT TO TRY AGAIN Y/N?"
120 INPUT D$
130 IF D$="Y" THEN 10
140 END
```

Try typing this program in your computer and run it. HAVE FUN!



Line 70 will check to see if the input is equal to the product.

If an incorrect product is input, the computer executes line 80 printing WRONG! TRY AGAIN. It then proceeds to line 50. A correct product will send the execution to line 100 printing CORRECT!. Lines 110-130 allow you to either run the program again or end it. (Recall the Y in line 130 is in quotes since it is being compared to an alphanumeric variable.)



SUGGESTED EXERCISES

1. Write a program that will print the same message continuously

until you break out of the program. (Make certain that you know how to break out of a program before you run it.)

2. Write a program that will print the first 22 multiples of 8.

3. Write a program that will ask the user to input two numbers and their sum. The program will then check to see if the user's sum is correct and print an appropriate message.

4. Write a program that requests the lengths of the three sides of any triangle. Have the program classify the triangle as scalene, isosceles, or equilateral.

5. Write a program that will ask for an angle (between 0 degrees and 180 degrees) and will classify it as acute, right, or obtuse.

6. Write a program that will find the measure of the 3rd angle of a triangle given the measures of the other two angles. Include a check so that the sum of the two given angles is less than 180 degrees.

7. Write a program that will output all the prime numbers from 10 to 100. (HINT: These will not be divisible by 2, 3, 5, or 7).

8. Fred goes shopping often. He must decide if it is cheaper to buy 5 cans of corn for \$.95 or 3 cans of corn for \$.54. Write a program to tell Fred which is cheaper per can. In your program, allow the prices of \$.95 and \$.54 to be variables so that these prices may change but still give Fred the correct answer.

9. The monthly rates charged by Super Shock Electric Co. are as follows (kwhr=kilowatt hours):

Base charge-----\$.95

First 100 kilowatt hours of use-----\$.09 per kwhr

Next 200 kwhr-----\$.08 per kwhr

Over 300 kwhr-----0.07 per kwhr

Given the number of kilowatt hours, write a program to print the total monthly charge for electricity.

4-10 GETTING LOOPED

OBJECTIVE: Upon completion of this section you should be able to write a program using loops. The program will use FOR-NEXT statements.

One of the major differences between a computer and a calculator is the computer's ability to do the same thing over and over an outrageous number of times. This is a loop. You can make loops with GOTO statements or FOR-NEXT statements.

Type in and run the following:

```
10 PRINT "MY COMPUTER IS OUT OF CONTROL"  
20 GOTO 10  
30 END
```

You will note this program runs continuously until you break out of it.

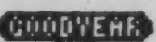
Now have the computer print the statement just ten times. We can use the FOR-NEXT statements as follows.

```
10 FOR N=1 TO 10  
20 PRINT "MY COMPUTER IS OUT OF CONTROL"  
30 NEXT N  
40 PRINT "AT LAST IT STOPPED!"  
50 END
```

As the computer begins, the value of N is 1. It prints MY COMPUTER IS OUT OF CONTROL the first time. Line 30 then increases N by one to 2, then goes to line 20 to print the message a second time. It repeats this pattern until the message is printed 10 times. Then it executes line 40 and line 50.

Note we can use variables other than N. Also, the variable does not need to start at 1. The following program will print the numbers from 40 to 55. TRY IT! 

```
10 FOR J=40 TO 55
20 PRINT J
30 NEXT J
40 END
```

Suppose you wanted to print all the EVEN numbers from 110 to 140. You can do this by using the word STEP in the statement FOR E=110 TO 140. The following program prints those even numbers. GIVE IT A TRY! 

```
10 FOR E=110 TO 140 STEP 2
20 PRINT E
30 NEXT E
40 END
```

STEP 2 indicates that E will be increased by 2 each time. Try the program again using STEP 5. This will give you the multiples of

5 from 110 to 140.

The FOR-NEXT statements are frequently used to make tables or charts as seen by the next program.

```

10 REM *****
20 REM **THIS PROGRAM GIVES A CHART FOR**
25 REM **THE SQUARE AND CUBE OF NUMBERS 1 THROUGH 10**
30 REM *****
40 PRINT TAB(3);"NUMBER";TAB(16);"SQUARE";TAB(25 );"CUBE"
50 FOR N=1 TO 10
60 PRINT TAB(3);N;TAB(16);N*N;TAB(25);N*N*N
70 NEXT N
80 END

```



Note that the headings are printed in line 40. This is BEFORE the beginning of the loop. Try this program for yourself. On a second run, try changing the tabs or change the numbers in line 50.

SUGGESTED EXERCISES

1. Write a program to list the numbers from 413 to 511.
2. Write a program to list all the multiples of 7 up to 700 (use STEP).
3. Write a program containing a loop that prints a message of your choice.

A) Use a GOTO statement.

B) Use FOR-NEXT statements to print the message 13 times.

4. Write a program that outputs a chart for the squares of the EVEN numbers from 20-30. The headings should be "THE NUMBER" and "THE SQUARE".
5. Write a program that outputs a chart telling the sums of the interior angles of polygons from 3 sides (triangle) to 10 sides (decagon). formula: $(N-2)*180$ where N=number of sides. The headings should be "NUMBER OF SIDES" and "SUM OF ANGLES".
6. The U.S. is exporting 30,000 bushels of wheat this year to Wheatheopia. Every year this will increase 5000 bushels. Write a program using FOR-NEXT statements to tell how many bushels the U.S. will export for each of the next 15 years.
7. The population of Portland increases by 300 people each year. This year the population is 21,800. Write a program using FOR-NEXT statements to show Portland's population in 2001.
8. Write a program that will print a temperature conversion table. Use DEGREES FAHRENHEIT and DEGREES CELSIUS as headings. Start at 0 degrees F and go to 212 degrees F in steps of 1. Use the equation $C=(F-32)/1.8$.
9. Write a program that will print a conversion table for any pair of English and Metric units. Pounds and Kilograms for example. Select beginning and ending points that seem useful.
10. Write a program that will print a multiplication table.

4-11 MEET THE ELECTRONIC FINGERS

OBJECTIVE: Upon completion of this section, you should be able to write a program using counters and/or adders.

In programs that contain loops, it may be desired to know how many times a loop is executed. A counter is used for this. A counter is a statement in which a variable is assigned a number (for example 0), and then each time the loop is performed, the variable is increased by a constant number (usually one).

The following program allows the operator to input a list of names. The counter will keep track of the number of names entered.

```

10 REM ++THIS PROGRAM IS A DEMONSTRATION OF A COUNTER++
20 LET C=0
30 PRINT "TYPE IN THE NAME"
40 PRINT TAB(15); "TYPE DONE AFTER LAST NAME"
50 INPUT N$
60 IF N$="DONE" THEN 100
70 REM++LINE 80 IS YOUR COUNTER STATEMENT++
80 LET C=C+1
90 GOTO 30
100 PRINT "THERE WERE ";C;" NAMES IN THE LIST."
110 END

```

Try this program a couple of times with different lists of names.



Note at the beginning of the program the variable C is equal to 0. After the first name is entered, line 80 increases C by 1. Line 90 instructs the computer to return to line 50. If a second name is entered, line 80 increases C by 1 again. The value of C then equals 2. This process will be repeated until DONE has been entered. Upon entering DONE, line 90 will instruct the computer to go to line 100 and complete the program. (Note: In line 80, "DONE" is the indicator that the input is complete. An indicator like this is called a flag.)

In a similar process, we can use an adder to continually add input to a given variable. In an adder statement, you will have:

 a variable = same variable + new variable (Ex. $P=P+Q$).

Notice how an adder is used in the next program.

```

10 REM ++THIS PROGRAM DEMONSTRATES AN ADDER++
20 LET A=0
30 PRINT "ENTER YOUR NUMBER."
40 PRINT "ENTER -1 AFTER LAST NUMBER."
50 INPUT X
55 REM ++FLAG IN LINE 60++
60 IF X=-1 THEN 90
65 REM ++LINE 70 IS YOUR ADDER STATEMENT++
70 LET A=A+X
80 GOTO 50

```



```

PAGE 56
90 PRINT "THE SUM OF THE NUMBERS IS ";A
100 END

```

When you run this program, try a few different lists of numbers. In the program, line 20 assigns A the value of 0. After the first number is entered, line 70 adds the value of X to A. Line 80 will instruct the computer to return to line 30. If a second number is entered, line 70 adds the value of X to the previous sum. (The program is keeping a 'running total'.) When -1 is entered, the number list is finished. Line 80 will check for this condition. When X=-1, the computer proceeds to line 90, completing the program. As more numbers are entered, each one is added to the previous sum.

The next program uses both a counter and adder to determine the average (mean) of a list of numbers.

```

10 REM**AVERAGING PROGRAM**
20 REM**N IS THE COUNTER**
30 LET N=0
40 REM**S IS THE ADDER**
50 LET S=0
60 PRINT "ENTER A NUMBER WHEN ? APPEARS"
70 PRINT "ENTER -1 AFTER LAST NUMBER"
80 INPUT D
90 IF D=-1 THEN 140
100 LET S=S+D
110 LET N=N+1

```

```

120 GOTO 80
130 REM**THE VARIABLE A IS THE AVERAGE**
140 LET A=S/N
145 PRINT
150 PRINT "THE NUMBER OF NUMBERS IN YOUR LIST WAS ";N
155 PRINT
160 PRINT "THEIR SUM WAS ";S
165 PRINT
170 PRINT "THE AVERAGE IS ";A
180 END

```



Remember that a statement number followed by only the word PRINT skips a line on the output.

Type in the AVERAGING PROGRAM and run it with the following lists of numbers:

(A) 70,85,80,75,80---Is your average 80?

(B) 278,283,308,288,313,284---Is your average 292?

SUGGESTED EXERCISES

1. Run the AVERAGING PROGRAM using the following sets of numbers:

(A) 1.7,2.7,4.3,8.5,6.3---ans. 4.7

(B) 0.23,2.3,10.08,0.567,3.88,0.8---ans. 2.9845

2. Write a program to add the whole numbers from 1 to 200.

3. Write a program to add the even whole numbers from 2 to 100.

4. Write a program to figure the following: $.5+1.0+1.5+\dots+9.5$

5. Write a program to find a student's average. The output should state the average. If the student has an average of 84.5 or above,

the output will state that this student is on the honor roll.

6. Hal has \$200 in the bank. Each 6 months, 5% interest is added to make a new amount. Write a program to tell how much money he will have after 5 years.

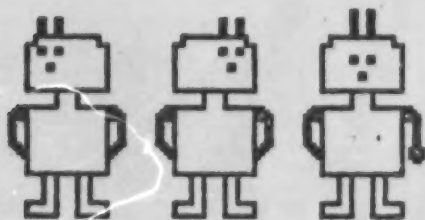
7. Jimmy The Rat ate 10 g of liver-flavored Cheetos on day one.

He ate 20 g of liver-flavored Cheetos on day two.

He ate 40 g of liver-flavored Cheetos on day three.

Each day, Jimmy continued to eat twice as much as he ate the previous day. Write a program to find out how much he ate just the tenth day and how much he totally ate during the ten days.

THE
AMAZING
MICROCOMPUTER



MEET THE AMAZING MICROCOMPUTER

During the 1970's, computers the size of typewriters were invented and developed. They are being used in businesses, schools, and homes. New uses are being discovered daily. This story tells how microcomputers are being used today. It tells what they can do and takes a look at what they might do in the future.

By using the BASIC Programming language, most people can use a micro to do simple calculations. A set of instructions called a program can be used over and over again. This makes the computer much more powerful than a simple calculator. Complicated calculations can also be performed easily and quickly. Computers can also make decisions. They can only make decisions that they are instructed to make, however. The person who writes the instructions is called a programmer. Many people are hired to program and maintain computers. They are employed by businesses, governments, and schools. They program microcomputers and the larger computers that are also used. The bigger computers may be called minicomputers or mainframes. They work faster and can handle more data. Microcomputers can also send information to bigger computers or to each other.

For many small businesses, microcomputers are used to keep tabs on raw materials and finished goods. Such a record is called an inventory. Micros can also print pay checks and store payroll data. They record the money owed to each supplier and the money owed by each customer. Other common tasks include the printing of mailing labels,

keeping records on employees, and word processing. Word processing is a lot like typing. The difference is that the word typed appear on a TV screen. The typed material is stored on a magnetic disk. This means that it can be printed over and over again. It can also be changed or corrected without retyping. This saves a lot of time for the typist. This text was prepared on a microcomputer using a word processing program. Even big businesses use microcomputers for word processing and for some calculations. They are portable and you can get your answers fast as long as you do not have to wait for someone else to run the micro for you.

At the heart of the microcomputer is a single chip called a microprocessor. This is where the instructions are carried out. Microprocessors are being used to run other machines. They are even being used in automobiles and cameras. Some factories use microprocessors in robots that assemble many products. The cost of these chips has decreased as they have become smaller. This has allowed for the production of microcomputers that cost less than \$1000. Many of these inexpensive models are sold for use in the home and in schools.

In addition to letters and numbers, micros can print other shapes on the screen. Such shapes are called graphics. Each dot on the screen can be turned on or off to form pictures on the screen. When shapes are drawn and erased and drawn again they appear to move. Moving shapes are used in most video games. These games can be run on many micros also. This added feature makes the micros more attractive

as products for the home. Some micros can also produce sound. You may be able to get one note at a time or you might be able to plug into a stereo and make electronic music.

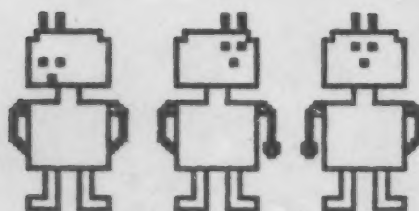
In schools the micros are being used to help teach subjects like math and reading. They are also used to teach programming and word processing. In some cases they are used to create student schedules and keep track of attendance. As more micros are purchased by school districts the number of uses will also increase.

In the future micros will become even more powerful. They will also be used for more things. You may have one in your home that will be used to do your banking and shopping. You can send messages to your friends and do your job without leaving home. You might ask for all of the news items that deal with a topic that you are interested in. You can enter your symptoms and get medical advice or you can create your own art and music. We are only limited by our own imaginations when we use this powerful mind tool.

NOTE TO INSTRUCTOR: See the disks provided with this package for programs that demonstrate the use of sound and graphics.

OTHER

GOODIES III



8-1 LIST OF BASIC COMMANDS - ALL COMMANDS MUST BE FOLLOWED BY PRESSING THE RETURN KEY

COMMAND	APPLE	TRS-80	HP 2000	PET
CLEAR SCREEN	HOME	CLS	SEE INSTRUCTOR	SHIFT/CLEAR HOME
CONTINUE PROGRAM	CONT	CONT	NO SCROLL 2X	CONT
DELETE CURRENT PROGRAM	NEW	NEW	SCRATCH	NEW
DELETE LINE X	LINE X	LINE X	LINE X	LINE X
DELETE LINES X THROUGH Y	DEL X,Y	NA	DEL-X,Y	NA
DELETE PROGRAM FROM DISK	DELETE NAME	NA	KILL-NAME	NA
DIVISION	/	/	/	/
EXPONENT	^	^	^	^
LIST ENTIRE PROGRAM	LIST	LIST	LIST	LIST
LIST FROM LINE X TO Y	LIST X,Y	LIST X,Y	LIST-X,Y	LIST X-Y
LIST ONLY LINE X	LIST X	LIST X	LIST-X,X	LIST X
LOAD PROGRAM	LOAD NAME	CLOAD "A"	GET-NAME	LOAD "NAME"
MULTIPLICATION	*	*	*	*
PRINTER OFF	PR#0	PRINT LPRINT	USE PRINTING TERMINAL	SEE INSTRUCTOR
PRINTER ON	PR#1		USE PRINTING TERMINAL	SEE INSTRUCTOR
RESET ALL VARIABLES TO ZERO	CLEAR	CLEAR	NA	CLEAR
RUN PROGRAM IN MEMORY	RUN	RUN	RUN	RUN
RUN PROGRAM FROM DISK	RUN NAME	NA	GET-NAME RUN	NA
SAVE PROGRAM	SAVE NAME	CSAVE "A"	SAVE-NAME	SAVE "NAME"
STOP A LISTING	CTRL C	BREAK	NO SCROLL	RUN/STOP
STOP A PROGRAM	CTRL C	BREAK	CTRL C	RUN/STOP
CATALOG	CATALOG	NA	CAT	NA

PREPARED SOFTWARE

THE FOLLOWING PROGRAMS ARE LOCATED ON TWO DISKS, IDENTIFIED AS D1 AND D2. D1 CONTAINS NUMBERS 1 THROUGH 22 OF THE FOLLOWING LIST. D2 CONTAINS NUMBERS 23 THROUGH 46. FOLLOWING EACH NUMBER IS A LETTER IN PARENTHESES. (G) INDICATES GAME. (D) INDICATES DEMO--THIS CAN BE SOUND AND/OR GRAPHICS. (I) INDICATES INSTRUCTIONAL.

THEY ARE A VARIETY OF PROGRAMS THAT YOU CAN LOOK AT, PLAY WITH, AND ENJOY WHILE LEARNING AT THE SAME TIME. A BRIEF EXPLANATION FOLLOWS EACH PROGRAM NAME.

1. (I) ATOMO - Questions on identification of symbols and numbers in the periodic table. Five (5) examples in each session. A periodic table should be available.
2. (D) RABBIT - Graphic illustration where the student must approximate distance. Time is given and student must determine velocity. Formula for velocity necessary ($v=d/t$). Two to five players.
3. (I) RESISTANCE - Graphic illustration of two circuits to find resistance. Involves formula for resistance, the addition of fractions with unlike denominators and the reciprocal of a fraction. After each problem, math is fully explained.
4. (I) BAR GRAPH - Makes a vertical bar graph up to 10 bars.
5. (G) GOLF - Metric game for calculating angles and distance. Angles can be entered as positive or negative values.
6. (D) LINE - Graphic illustrations of lines in the form of $y = mx + b$. Students enter slope (in decimal form) and y - intercept. Used for grade 9.
7. (I) MATHRACE - Race game involving multiplication or division using significant digits without rounding off. Two players.
8. (I) GRADUATE - Drill on reading volumes in a graduated cylinder. Must determine increments.
9. (I) METRICK - Approximating distances using various metric units.
10. (D) TIC-TAC-TOE - Game of graphics involving two people. Requires nine multiple choice questions (4 answers), on any topic.
11. (G) MINIGOLF - Practice in angles for any level. Using arrows-locate the club in a circle around the ball. Press any number from 1 to 9. The higher the number, the farther the ball moves. One to five players.

12. (G) FLAGS - ** OPERATES ONLY WITH PADDLES ** - Social Studies. Color TV required. Student can identify up to 48 national flags.
13. (I) HEXADECIMAL PUZZLE - Rearranging hexadecimal values in ascending order.
14. (G) OLD GLORY - Illustrates United States flags from 1777 to present. Star- Spangled Banner is played. Color TV preferred.
15. (I) MATH DECATHLON - This is a game of mathematical skills and reasoning for one to four players. There are ten separate events. There are different skill levels.
16. (G) FOOTBALL - Game for three choices of competition. All levels.
17. (I) BONE TUMOR DIAGNOSIS - Program for physicians only. Determines prognosis of tumor of the bone.
18. (I) CORONARY RISK - This program will calculate your life span using coronary risk factors.
19. (G) LEMONADE - Selling lemonade at a stand, depending on cost, weather and other factors.
20. (I) NUKE - Game to simulate the operation of a nuclear power reactor. The object is to operate the plant at a maximum average power input without causing a reactor meltdown.
21. (I) HABER - Chemistry program which enables you to study the reversible reaction which occurs in the Haber synthesis of ammonia.
22. (I) PLANETARY ORBITS - Graphically illustrates orbits of the nine planets. Shows weekly or monthly markers. Also shows Haley's comet.
23. (I) CIVIL WAR- Game is based on 14 battles of the Civil War. Facts and figures are based on actual occurrences.
24. (G) OLD DUTCHMAN'S GOLD - Search for the Old Dutchman's gold by using clues that you find in the program.
25. (I) PRIME FACTORS - Finds prime factors of numbers.
26. (D) LOW RESOLUTION DEMO - Illustrates ball going from upper left corner to lower right corner of the screen. Program can be altered to change direction of the ball.
27. (I) WIND CHILL - Determines wind chill factor.

28. (C) NIMBOT - Game using three rows of robots in which the player shooting the last robot loses.

29. (D) LOW RESOLUTION DEMO-NUKE - Illustrates the Apple's low resolution graphic capability.

30. (I) QUIZ MODEL - The purpose of this program is to allow an inexperienced programmer to create quizzes that will run on a microcomputer. See separate instructions.

31. (D) SOUND DEMO - Demonstrates all possible pitches that can be produced on an Apple.

32. (D) XMAS TREE - Demonstrates simple, low resolution graphics.

* * * * * INTEGER PROGRAMS * * * * *

If you catalog the following programs, you will notice that the titles are prefaced by an I. This means that they are written in a different form of Basic, called Integer Basic.

Before these programs can be run, you must first command RUN INTEGER. After doing that, run the program as usual.

33. INTEGER - A program to convert from Applesoft to Integer Basic.

34. (C) TWENTY THREE BRICKS - Game in which the object is to avoid taking the last brick. Needs paddles.

35. (C) SEVEN - Primarily a version of Crazy Eights. The object of the game is to get rid of all your cards by following suit or rank of the 'show' card.

36. (I) COLOR MATH - Simple addition, subtraction, multiplication and division problems. Problem is done until correct answer is achieved.

37. (D) ALMA MATER - Simple demonstration of how the Apple makes music.

38. (D) CURVES - Graphic illustration.

39. (C) TOWERS OF HANOI - Game for one player in which player transfers discs among three posts.

40. (D) MORSE CODE - Morse code translator with varying speeds.

- 41. (C) BOWLING - Bowling game for 1 or 2 players.
- 42. (C) HAMMURABI - Came to govern village by determining food requirements (i.e. buying, selling, planting).
- 43. (D) APPLEVISION - Graphic demonstration with music.
- 44. (I) SUPERMATH - Excellent exercise in addition, subtraction, multiplication and division. You can either choose or achieve level of difficulty. Good explanation given for incorrect answers.
- 45. (I) BIORHYTHM - Predicts critical days for physical, emotional and mental status for individuals.
- 46. (D) MAZE - Getting through a three dimensional maze at a selected level of difficulty.

```
5 REM +++SECTION 2-1+++  
10 PRINT "THIS IS EASY"  
20 END
```

```
5 REM +++SECTION 2-2+++  
10 PRINT "JOE JONES"  
20 PRINT "6 ALISON LANE"  
30 PRINT "MARATHON, NEW YORK"  
40 PRINT "19803"  
50 END
```

```
5 REM +++SECTION 2-3+++  
10 PRINT "NOW IS THE TIME FOR ALL GOOD MEN"  
20 PRINT "TO COME TO THE AID OF THEIR COUNTRY"  
30 END
```

```
5 REM +++SECTION 3-1+++  
10 PRINT 15 + 3  
20 PRINT 15 - 3  
30 PRINT 15 * 3  
40 PRINT 15 / 3  
50 END
```

```
5 REM +++SECTION 3-2+++  
10 PRINT 8 ^ 2  
20 END
```

```
5 REM +++SECTION 3-3+++
10 PRINT "(16+4)-(4-2)"
20 PRINT (16 + 4) - (4 - 2)
30 PRINT "(16/4)^2"
40 PRINT (16 / 4) ^ 2
50 END
```

```
5 REM +++SECTION 4-1+++
10 LET Q = 28
20 LET R = 36
30 PRINT "THE SUM OF ";Q;" AND ";R;" IS ";Q + R
40 END
```

```
5 REM +++SECTION 4-2+++
10 LET T = 985
20 LET L = 637
30 PRINT "THE PRODUCT OF ";T;" AND ";L;" IS ";T * L
40 END
```

```
5 REM +++SECTION 5-1+++
10 PRINT "(8*12/3)-(2*5)"
20 PRINT (8 * 12 / 3) - (2 * 5)
30 END
```

```
5 REM +++SECTION 5-2+++
10 PRINT "8*12/(3-2)*5"
20 PRINT 8 * 12 / (3 - 2) * 5
30 END
```

```
5  REM +++SECTION 5-3+++
10  PRINT "5*(24-2)/(1+1)"
20  PRINT 5 * (24 - 2) / (1 + 1)
30  END
```

```
5  REM +++SECTION 5-4+++
10  PRINT "96/(24/2)"
20  PRINT 96 / (24 / 2)
30  PRINT "(96/24)/2"
40  PRINT (96 / 24) / 2
50  END
```

```
5  REM +++SECTION 5-5+++
10  PRINT "10-2^3"
20  PRINT 10 - 2 ^ 3
30  PRINT "(10-2)^3"
40  PRINT (10 - 2) ^ 3
50  END
```

```
5  REM SECTION+++5-6+++
10  PRINT "8-2-(2-1)"
20  PRINT 8 - 2 - (2 - 1)
30  PRINT "8-(2-2)-1"
40  PRINT 8 - (2 - 2) - 1
50  END
```



```

5  REM  SECTION+++5-7+++
10  PRINT "(4+4)-(4+4)"
20  PRINT (4 + 4) - (4 + 4)
30  PRINT " 4+(4/4)-4"
31  PRINT 4 + (4 / 4) - 4
32  PRINT "(4/4) +(4/4)"
38  PRINT (4 / 4) + (4 / 4)
50  PRINT "(4+4+4)/4"
60  PRINT (4 + 4 + 4) / 4
70  PRINT "((4-4)/4) + 4"
80  PRINT ((4 - 4) / 4) + 4
90  PRINT "((4*4)+4)/4"
100 PRINT ((4 * 4) + 4) / 4
110 PRINT "((4+4)/4)+4"
120 PRINT ((4 + 4) / 4) + 4
130 PRINT " (4+4)-4/4"
140 PRINT (4 + 4) - 4 / 4
150 PRINT " (4+4+4)-4"
160 PRINT (4 + 4 + 4) - 4
170 PRINT " (4+4)+4/4"
180 PRINT (4 + 4) + 4 / 4
200  END

```

```

5  REM  SECTION +++8-1+++
10  PRINT "GIVEN THE LENGTH OF THE SIDE OF A SQUARE, THIS PROGRAM WILL FIND THE PERIMETER."
20  INPUT S
30  LET P = 4 * S
40  PRINT "THE PERIMETER OF THE SQUARE IS ";P
50  END

```

```

5  REM  SECTION+++8-2+++
10  PRINT "BILL'S SALARY FOR HIS WORK IS TO BE COMPUTED"
20  INPUT H
30  LET S = H * 3.50
40  PRINT "BILL'S SALARY IS ";S
50  END

```

```

5  REM  SECTION+++8-3+++
10  PRINT "THIS PROGRAM WILL ASK FOR THE AREA OF A RECTANGLE"
20  PRINT "THE LENGTH"
30  INPUT L
40  PRINT "THE WIDTH"
50  INPUT W
60  LET A = L * W
70  PRINT "THE AREA IS",A
80  END

```

```

5  REM  SECTION+++8-4+++
10  PRINT "THIS PROGRAM WILL CALCULATE YOUR AVERAGE GRADE"
15  PRINT "TYPE IN ANY SIX GRADES"
20  INPUT A,B,C,D,E,F
30  LET A = (A + B + C + D + E + F) / 6
40  PRINT "THE AVERAGE OF YOUR MARKS IS      ";A
50  END

```

```

5  REM  SECTION+++8-5A+++
10  PRINT "THIS WIL CONVERT DEGREES CELSIUS TO DEGREES FAHRENHEIT FOR YOU"
20  INPUT C
30  LET F = C * (9 / 5) + 32
40  PRINT "GIVEN THE TEMPERATURE IN DEGREES CELSIUS, THE TEMPERATURE IN DE
      GREES FAHRENHEIT WOULD BE      ";F
50  END

```

```

5  REM  SECTION+++8-5B+++
10  PRINT "THIS PROGRAM WILL CONVERT DEGREES FAHRENHEIT TO DEGREES CELSIUS
      "
20  INPUT F
30  LET C = (F - 32) * (5 / 9)
40  PRINT "GIVEN THE TEMPERATURE IN DEGREES FAHRENHEIT, THE TEMPERATURE IN
      DEGREES CELSIUS IS      ";C
50  END

```

```

5  REM  SECTION+++6-6+++
10 PRINT "THIS PROGRAM WILL CALCULATE THE MILES PER GALLON FOR A CAR"
20 PRINT "HOW MANY MILES?"
30 INPUT M
40 PRINT "HOW MANY GALLONS?"
50 INPUT G
60 LET Z = M / G
70 PRINT "YOUR MILES PER GALLON IS ";Z
80 END

```

```

5  REM  SECTION +++6-7+++
10 PRINT "THIS PROGRAM GIVES TWO ODOMETER READINGS FOR A CAR AND THE NUMB
    ER OF GALLONS USED"
20 PRINT "FIRST ODOMETER READING?"
30 INPUT F
40 PRINT "LAST ODOMETER READING?"
50 INPUT L
60 PRINT "GALLONS OF GAS USED?"
70 INPUT G
80 LET M = (L - F) / G
90 PRINT "THE MILES PER GALLON IS ";M
100 END

```

```

5  REM  SECTION+++6-8+++
10 PRINT " THIS PROGRAM TELLS YOU THE DISCOUNT AND THE PRICE YOU WOULD PA
    Y FOR AN ITEM AT A PERCENT OF DISCOUNT"
15 PRINT "ENTER THE ORIGINAL COST ,THEN THE PERCENT DISCOUNT"
20 INPUT X,Y
30 LET D = X * (.01) * Y
40 LET C = X - (X * (.01) * Y)
50 PRINT "THE DISCOUNT ALLOWED ON THE ITEM IS ";D
60 PRINT "THE FINAL SALES PRICE ON THE ITEM IS ";C
70 END

```

```

10 REM +++SECTION 7-1+++
20 LET A$ = "DEAR JOHN"
30 LET B$ = "      I WENT SHOPPING.  BE BACK IN A      LITTLE WHILE."
40 LET C$ = "      MARY"
50 HOME
60 PRINT
70 PRINT A$
80 PRINT
90 PRINT B$
100 PRINT
110 PRINT C$
120 END

```

```

10 REM +++SECTION 7-2+++
20 PRINT
30 PRINT "ENTER NAME"
40 INPUT NA$
50 HOME
60 PRINT NA$; ", "
70 PRINT
80 PRINT "THIS IS A MESSAGE!"
90 END

```

```

10 REM +++SECTION 7-3+++
20 PRINT "THE DATE"
30 INPUT D$
40 PRINT "YOUR TEACHER"
50 INPUT T$
60 PRINT "THE REASON"
70 INPUT R$
80 HOME
90 PRINT D$
100 PRINT
110 PRINT
120 PRINT "DEAR "T$
130 PRINT
140 PRINT "I AM SORRY THAT I DID NOT HAND IN MY      HOMEWORK.  THE REASON I
    S:"
150 PRINT R$
160 PRINT
170 PRINT "A STUDENT"
180 END

```



```
10 REM +++SECTION 7-4+++
20 PRINT "NAME OF CONTRACTOR"
30 INPUT C$
40 PRINT "THE DATE"
50 INPUT D$
60 PRINT "WHAT IS IN THE SHIPMENT?"
70 INPUT S$
80 PRINT "THE COST"
90 INPUT CO$
100 HOME
110 PRINT D$
120 PRINT
130 PRINT "TO ";C$
140 PRINT
150 PRINT "YOUR SHIPMENT HAS ARRIVED. IT CONSISTS OF THE FOLLOWING:"
160 PRINT S$
170 PRINT "YOU OWE US "CO$
180 PRINT
190 PRINT "NO-VALUE HARDWARE STORE"
200 END
```

```

10 REM +++SECTION 7-5+++
20 PRINT "GUEST'S NAME"
30 INPUT G$
40 PRINT "GUEST'S ADDRESS"
50 INPUT A$
60 HOME
70 PRINT
80 PRINT "DEAR ";G$
90 PRINT
100 PRINT "    I WANT YOU TO COME TO MY PARTY THIS FRIDAY.  IT IS AT MY H
    OUSE AND STARTS    AT 4, RIGHT AFTER SCHOOL.  THE PARTY    IS A SURPRI
    SE FOR MALTILDA'S BIRTHDAY SO    BRING A PRESENT.  SEE YOU THEN."
110 PRINT
120 PRINT "JENNY PARTY"
130 PRINT
140 PRINT "MOM, PLEASE MAIL TO ";G$
150 PRINT "THE ADDRESS IS ";A$
160 END

```

```

5 REM +++SECTION 8-1+++
10 PRINT "THIS PROGRAM USES THE TAB TO PRINT A NAME"
15 PRINT
20 PRINT TAB( 17);"NORMA PARKER"
25 END

```

```

9 REM +++SECTION 8-2+++
9 PRINT "THIS PROGRAM WILL TAB TWO NUMBERS AND THEIR PRODUCTS"
10 LET X = 5
20 LET Y = 7
30 LET Z = X * Y
40 PRINT TAB( 7);X; TAB( 17);Y; TAB( 27);Z
50 END

```

```

5 REM +++SECTION 8-3+++
10 PRINT "THIS PROGRAM USES A TAB TO REPRODUCE A DESIGN"
20 PRINT TAB( 10);"@"
30 PRINT TAB( 10);"####"
40 PRINT TAB( 17);"#####"
50 PRINT TAB( 16);"#####"
60 PRINT TAB( 15);"#####"
70 PRINT TAB( 16);"#####"
80 PRINT TAB( 17);"#####"
90 PRINT TAB( 18);"####"
100 PRINT TAB( 10);"@"
110 END

```

```

5 REM +++SECTION 8-4+++
10 PRINT "THIS PROGRAM PRINTS INITIALS ON THE SCREEN"
20 PRINT TAB( 8);"@"; TAB( 12);"@"; TAB( 28);"####"
30 PRINT TAB( 8);"@"; TAB( 9);"@"; TAB( 12);"@"; TAB( 28);"@"; TAB( 32);
   "@@"
40 PRINT TAB( 8);"@"; TAB( 10);"@"; TAB( 12);"@"; TAB( 28);"####"
50 PRINT TAB( 8);"@"; TAB( 11);"@"; TAB( 12);"@"; TAB( 28);"@"
60 PRINT TAB( 8);"@"; TAB( 12);"@"; TAB( 28);"@"
100 END

```

```

5 REM +++SECTION 8-5+++
10 PRINT "THIS PROGRAM PRINTS A DESIGN ON THE SCREEN"
20 PRINT TAB( 17);"*****"
30 PRINT TAB( 17);"***"; TAB( 21);"***"
40 PRINT TAB( 17);"***"; TAB( 22);"***"
50 PRINT TAB( 17);"***"; TAB( 19);"***"; TAB( 22);"***"
60 PRINT TAB( 17);"***"; TAB( 22);"***"
70 PRINT TAB( 17);"***"; TAB( 21);"***"
80 PRINT TAB( 17);"*****"
100 END

```

```

3  REM ++++SECTION 6-6++++
10 PRINT "THIS PROGRAM ANNOUNCES THE BIRTH OF A BABY"
20 LET S$ = "WE HAVE A SON!"
30 LET T$ = "WE LISTED HIM AS"
40 LET X$ = "SAM, JR."
50 LET Z$ = "AND DATED HIM ON"
60 LET Q$ = "JUNE 27, 1981"
60 PRINT TAB( 12);S$
65 PRINT
60 PRINT TAB( 12);T$
65 PRINT
100 PRINT TAB( 17);X$
105 PRINT
110 PRINT TAB( 12);Z$
115 PRINT
120 PRINT TAB( 13);Q$
150 END

```

```

3  REM +++SECTION 6-7+++
8  HOME
10 PRINT "THIS PROGRAM DESIGNS A LETTER TO BE SENT TO SEASON 'FOOTBRAWL'
    TICKET HOLDERS, TELLING THEM HOW MUCH THEY STILL OWE ON THEIR TICKETS
    "
15 PRINT
100 PRINT "DATE ";
110 INPUT D$
120 PRINT
130 PRINT "SEAT LOCATION ";
140 INPUT SL$
150 PRINT
160 PRINT "MONEY OWED ";
170 INPUT MO$
180 HOME
200 PRINT TAB( 25);D$
210 PRINT
250 PRINT
260 PRINT "DEAR SEASON TICKET HOLDER,"
270 PRINT
280 PRINT TAB( 5);"GLAD YOU ARE COMING TO THE OPENING GAME ON SEPTEMBER
    10, 1981. AS YOU KNOW, OUR OPPONENT IS SYRACUSE"
285 PRINT
290 PRINT TAB( 5);"YOUR SEAT LOCATION WILL BE:"
300 PRINT
310 PRINT TAB( 5);SL$
320 PRINT
330 PRINT TAB( 5);"THE BALANCE REMAINING IS $"MO$
340 PRINT
350 PRINT TAB( 5);"THANK YOU FOR YOUR SUPPORT."
360 PRINT
370 PRINT TAB( 25);"SINCERELY,"
380 PRINT : PRINT
390 PRINT TAB( 25);"EAGAR T. ALUMNI"

```



```

5 REM +++SECTION 9-1+++
10 PRINT "THIS PROGRAM PRINTS A MESSAGE CONTINUOUSLY UNTIL YOU BREAK OUT
    OF THE PROGRAM"
20 PRINT "WHAT IS THE WEATHER TODAY?"
30 GOTO 10
40 END

```

```

5 REM +++SECTION 9-2+++
10 PRINT "THIS PROGRAM GIVES THE MULTIPLES OF 8 THROUGH THE FIRST 22 MUL
    TIPLES"
20 LET N = 8
25 LET R = 1
30 PRINT N,N * R
35 IF R = 22 THEN 100
55 LET R = R + 1
65 REM THE VALUE OF R IS INCREASED BY 1
70 GOTO 30
100 END

```

```

5 REM +++SECTION 9-3+++
10 PRINT "THIS PROGRAM CHECKS YOUR ABILITY TO ADD"
15 PRINT
20 PRINT "LIST ANY TWO NUMBERS"
22 PRINT
25 INPUT A,B
27 PRINT
30 PRINT "WHAT IS THE SUM OF THESE NUMBERS?"
32 PRINT
33 LET D = A + B
34 PRINT
35 INPUT C
38 IF D < > C THEN 70
45 IF D = C THEN PRINT "GOOD"
48 GOTO 78
50 IF D < > C THEN 70
70 PRINT "YOUR ANSWER IS NOT CORRECT, TRY AGAIN"
72 PRINT
75 GOTO 30
78 PRINT
82 END

```

```

5  REM +++SECTION 9-4+++
10 PRINT "THIS PROGRAM HELPS YOU IDENTIFY DIFFERENT TYPES OF TRIANGLES"
15 PRINT
20 PRINT " A SCALENE TRIANGLE HAS THREE SIDES OF DIFFERING LENGTH"
25 PRINT
30 PRINT "AN ISOSCELES TRIANGLE HAS TWO SIDES OF THE SAME LENGTH AND THE
    THIRD SIDE OF A DIFFERENT LENGTH"
35 PRINT
40 PRINT "AN EQUILATERAL TRIANGLE HAS ALL SIDES OF THE SAME LENGTH"
45 PRINT
47 PRINT "WHAT ARE THE THREE SIDES?"
50 INPUT A,B,C
60 IF A = B THEN 110
70 IF B = C THEN 120
80 IF A = C THEN 120
90 PRINT "THE TRIANGLE IS SCALENE"
100 END
110 IF B = C THEN 140
120 PRINT "THE TRIANGLE IS ISOSCELES"
130 END
140 PRINT "THE TRIANGLE IS EQUILATERAL"
150 END

```

```

5  REM +++SECTION 9-5+++
10 PRINT " THIS PROGRAM IDENTIFIES ANGLES AS ACUTE, OBTUSE AND RIGHT"
20 PRINT "CHOOSE ANY ANGLE"
30 INPUT C
40 IF C < > 90 THEN 70
50 PRINT "IT IS A RIGHT ANGLE"
60 END
70 IF C < 0 THEN 130
80 IF C > 180 THEN 130
90 IF C < 90 THEN 120
100 PRINT "THE ANGLE IS OBTUSE"
110 END
120 PRINT "THE ANGLE IS ACUTE"
125 END
130 PRINT "THE ANGLE IS BETWEEN 0 AND 180"
140 PRINT "TRY AGAIN"
150 GOTO 30

```

```

10 REM +++SECTION 9-6++++
20 PRINT "WHAT IS THE FIRST ANGLE?"
30 INPUT X
40 PRINT "WHAT IS THE SECOND ANGLE?"
50 INPUT Y
60 LET Z = X + Y
70 IF Z >= 180 THEN 110
80 LET A = 180 - Z
90 PRINT "THE THIRD ANGLE IS ";A
100 GOTO 120
110 PRINT "IMPOSSIBLE! THE TWO ANGLES CAN NOT ADD UP TO 180 OR MORE."
120 END

```

```

10 REM SECTION 9-7
20 PRINT "THE FOLLOWING ARE THE PRIMES FROM 10 TO 100:"
30 FOR N = 10 TO 100
40 FOR A = 1 TO 50
50 IF N = (2 * A) THEN 110
60 IF N = (3 * A) THEN 110
70 IF N = (5 * A) THEN 110
80 IF N = (7 * A) THEN 110
90 NEXT A
100 PRINT N
110 NEXT N
120 END

```

```

5 REM +++SECTION 9-8+++
10 PRINT "THIS PROGRAM IS TO CALCULATE WAYS TO SAVE MONEY WHILE SHOPPING
    BY CALCULATING PRICES FOR MULTIPLE QUANTITIES"
15 PRINT
20 REM F WILL BE THE COST PER CAN WHEN BOUGHT IN LOTS OF FIVE
40 REM T WILL BE THE COST PER CAN WHEN BOUGHT IN LOTS OF THREE
50 PRINT "WHAT IS THE COST OF FIVE CANS OF FOOD?"
52 PRINT
55 INPUT Z
58 LET F = Z / 5
60 PRINT "WHAT IS THE COST OF THREE CANS OF FOOD?"
62 PRINT
65 INPUT D
68 LET T = D / 3
70 PRINT "THE COST PER CAN OF FOOD IN LOTS OF FIVE IS $ ";F
72 PRINT
75 PRINT "THE COST PER CAN OF FOOD IN LOTS OF THREE IS $ ";T
77 PRINT
80 PRINT "THE COST OF THE CANS WHEN BOUGHT IN GROUPS OF THREE IS CHEAPER"
85 END

```

```

10 REM +++SECTION 10-1+++
20 REM +++THIS PROGRAM PRINTS A LIST OF ALL NUMBERS FROM 423 TO 511+++
30 FOR A = 423 TO 511
40 PRINT A
50 NEXT A
60 END

```

```

10 REM +++SECTION 10-2+++
20 REM +++PROGRAM TO LIST ALL MULTIPLES OF 7 UP TO 700+++
30 FOR J = 7 TO 700 STEP 7
40 PRINT J
50 NEXT J
60 END

```

```

10 REM +++SECTION 10-3A+++
20 REM +++LOOP USING GOTO+++
30 PRINT "WILL THIS EVER STOP?"
40 GOTO 30
50 END

```

```

10 REM +++SECTION 10-3B+++
20 REM +++LOOP DONE 13 TIMES USING FOR-NEXT+++
30 FOR H = 1 TO 13
40 PRINT "I WILL PRINT THIS 13 TIMES."
50 NEXT H
60 END

```

```

10 REM +++SECTION 10-4+++
20 REM +++CHART FOR SQUARES OF EVEN NUMBERS FROM 20 TO 50+++
30 PRINT TAB( 10);"THE NUMBER"; TAB( 20);"THE SQUARE"
40 FOR R = 20 TO 50 STEP 2
50 PRINT TAB( 14);R; TAB( 32);R * R
60 NEXT R
70 END

```



```

10 REM +++SECTION 10-5+++
20 REM +++CHART TELLING SUMS OF INTERIOR ANGLES OF POLYGONS FROM TRIANGLE TO DECAGON+++
30 PRINT "NUMBER OF SIDES"; TAB( 20);"SUM OF ANGLES"
40 FOR N = 3 TO 10
50 LET S = (N - 2) * 180
60 PRINT TAB( 8);N; TAB( 32);S
70 NEXT N
80 END

```

```

10 REM +++SECTION 10-6+++
20 REM +++WHEAT EXPORTS FOR NEXT 15 YEARS+++
30 PRINT "U.S. WHEAT EXPORTS TO WHEATHEOPIA"
32 PRINT
33 PRINT
40 PRINT "YEAR"; TAB( 12);"BUSHEL OF WHEAT EXPORTED"
45 LET W = 30000
50 PRINT "1981"; TAB( 20);W
60 FOR Y = 1982 TO 1996
70 LET W = W + 5000
80 PRINT Y; TAB( 20);W
90 NEXT Y
100 END

```

```

10 REM +++SECTION 10-7+++
20 REM +++PROGRAM TO SHOW POPULATION UNTIL 2001 FOR CITY OF FORTLAND+++
30 PRINT "POPULATION OF CITY OF FORTLAND"
32 PRINT
40 PRINT "YEAR" TAB( 15);"PROJECTED POPULATION"
45 LET P = 21600
50 PRINT "1981"; TAB( 22);P
60 FOR Y = 1982 TO 2001
70 LET P = P + 300
80 PRINT Y; TAB( 22);P
90 NEXT Y
100 END

```

```

10 REM +++SECTION 10-8+++
20 REM +++PROGRAM TO PRINT TEMPERATURE CONVERSION TABLE+++
30 PRINT "DEGREES FAHRENHEIT"; TAB( 25);"DEGREES CELSIUS"
40 FOR F = 0 TO 212
50 LET C = (F - 32) / 1.8
60 PRINT TAB( 8);F; TAB( 31);C
70 NEXT F
80 END

```

```

10 REM +++SECTION 10-9+++
20 REM +++CONVERSION TABLE FOR POUNDS AND KILOGRAMS FROM 0 POUNDS TO 20 P
   OUNDS STEP .5+++
30 PRINT "POUNDS"; TAB( 20);"KILOGRAMS"
40 FOR P = 0 TO 20 STEP .5
50 LET K = P / 2.2
60 PRINT TAB( 3);P; TAB( 24);K
70 NEXT P
80 END

```

```

10 REM +++SECTION 10-10+++
20 REM +++PROGRAM TO PRINT MULTIPLICATION TABLE+++
30 PRINT "MULTIPLICATION TABLE"
40 PRINT
50 FOR T = 0 TO 10
60 PRINT T * 0; TAB( 3);T * 1; TAB( 6);T * 2; TAB( 9);T * 3; TAB( 12);T *
   4; TAB( 15);T * 5; TAB( 18);T * 6; TAB( 21);T * 7; TAB( 24);T * 8; TAB(
   27);T * 9; TAB( 30);T * 10
70 NEXT T
80 END

```

```

10 REM +++SECTION 11-1A+++
20 REM +++USE AVERAGING PROGRAM PROVIDED IN TEXT FOR 11-1A AND 11-1B+++
30 REM +++N IS COUNTER+++
40 LET N = 0
50 REM +++S IS ADDER+++
60 LET S = 0
70 PRINT "ENTER A NUMBER WHEN ? APPEARS"
80 PRINT "ENTER -1 AFTER LAST NUMBER"
90 INPUT D
100 IF D = - 1 THEN 150
110 LET N = N + 1
120 LET S = S + D
130 GOTO 90
140 REM +++VARIABLE A IS THE AVERAGE+++
150 LET A = S / N
155 PRINT
160 PRINT "THE NUMBER OF NUMBERS IN YOUR LIST IS ";N
165 PRINT
170 PRINT "THEIR SUM IS ";S
175 PRINT
180 PRINT "THE AVERAGE IS ";A
190 END

```

```

10 REM +++SECTION 11-1B+++
20 REM +++USE AVERAGING PROGRAM PROVIDED IN TEXT FOR 11-1A AND 11-1B+++
30 REM +++N IS COUNTER+++
40 LET N = 0
50 REM +++S IS ADDER+++
60 LET S = 0
70 PRINT "ENTER A NUMBER WHEN ? APPEARS"
80 PRINT "ENTER -1 AFTER LAST NUMBER"
90 INPUT D
100 IF D = - 1 THEN 150
110 LET N = N + 1
120 LET S = S + D
130 GOTO 90
140 REM +++VARIABLE A IS THE AVERAGE+++
150 LET A = S / N
155 PRINT
160 PRINT "THE NUMBER OF NUMBERS IN YOUR LIST IS ";N
165 PRINT
170 PRINT "THEIR SUM IS ";S
175 PRINT
180 PRINT "THE AVERAGE IS ";A
190 END

```

```

10 REM +++SECTION 11-2+++
20 REM +++PROGRAM TO ADD WHOLE NUMBERS FROM 1 TO 200+++
25 LET T = 0
30 FOR N = 1 TO 200
35 REM ++T IS ADDER++
40 LET T = T + N
50 NEXT N
60 PRINT "THE SUM OF THE WHOLE NUMBERS FROM 1 TO 200 IS ";T
70 END

```

```

10 REM +++SECTION 11-3+++
20 REM +++PROGRAM TO ADD EVEN WHOLE NUMBERS FROM 2 TO 200+++
30 LET A = 0
40 FOR N = 2 TO 200 STEP 2
45 REM +++A IS ADDER+++
50 LET A = A + N
60 NEXT N
70 PRINT "THE SUM OF THE EVEN WHOLE NUMBERS FROM 2 TO 200 IS ";A
80 END

```

```

10 REM +++SECTION 11-4+++
20 REM +++PROGRAM TO ADD .5+1.0+1.5...+9.0+9.5+++
30 LET T = 0
40 FOR N = .5 TO 9.5 STEP .5
45 REM +++T IS ADDER+++
50 LET T = T + N
60 NEXT N
70 PRINT "THE SUM IS ";T
80 END

```

```

5 REM +++SECTION 11-5+++
10 PRINT "THIS PROGRAM CALCULATES STUDENT AVERAGES AND IDENTIFIES THOSE O
    N THE HONOR ROLL"
12 PRINT
13 REM N IS THE COUNTER
20 LET N = 0
25 REM S IS THE ADDER
30 LET S = 0
40 PRINT "ENTER A GRADE WHEN ? APPEARS"
45 PRINT
50 PRINT "ENTER -1 AFTER LAST NUMBER"
55 PRINT
58 PRINT "WHAT ARE YOUR GRADES? "
60 INPUT D
70 IF D = - 1 THEN 120
80 LET S = S + D
90 LET N = N + 1
100 GOTO 60
110 REM ** THE VARIABLE A IS THE AVERAGE
120 LET A = S / N
125 PRINT
130 PRINT "THE NUMBER OF GRADES YOU HAD WAS ";N
135 PRINT
140 PRINT "THEIR SUM WAS ";S
145 PRINT
150 PRINT "YOUR AVERAGE IS ";A
155 PRINT
160 IF A = > 84.5 THEN 180
170 GOTO 180
180 PRINT "YOU ARE ON THE HONOR ROLL!"
190 END

```



```
10 REM +++SECTION 11-6+++
20 REM +++CALCULATE INTEREST SEMI-ANNUALLY WITH STARTING AMOUNT OF $200 W
   ITH 5% INTEREST EVERY 6 MONTHS+++
25 LET A = 200
30 FOR Y = 1 TO 5 STEP .5
40 LET Q = .05 * A
50 LET A = A + Q
60 NEXT Y
70 PRINT "THE AMOUNT OF MONEY AT THE END OF 5 YEARS WILL BE $";A
80 END
```

!SYNTAX ERROR

```
10 REM +++SECTION 11-7+++
20 REM +++JIMMY THE RAT PROBLEM+++
30 LET A = 10
35 LET Z = 10
40 FOR D = 2 TO 10
50 LET A = A * 2
55 REM +++Z IS ADDER+++
60 LET Z = Z + A
70 NEXT D
80 PRINT " ON DAY 10, JIMMY ATE ";A;" GRAMS OF LIVER-FLAVORED CHEETOS"
90 PRINT " JIMMY ATE A TOTAL OF ";Z;" GRAMS OF LIVER-FLAVORED CHEETOS"
100 END
```

QUIZ 1

1. THE CORRECT WORD TO ERASE THE CURRENT PROGRAM FROM THE COMPUTER'S MEMORY ON THE APPLE OR THE TRS-80 IS ---.

ANSWER QUESTION #1.....NEW

2. THE CORRECT WORD TO EXECUTE THE PROGRAM IN THE COMPUTER'S MEMORY IS ---.

ANSWER QUESTION #2.....RUN

3. THE CORRECT WORD TO PRINT THE LINES OF THE PROGRAM IN THE COMPUTER'S MEMORY IS ----.

ANSWER QUESTION #3.....LIST

4. AFTER EVERY COMMAND YOU MUST HIT THE ENTER OR ----- KEY.

ANSWER QUESTION #4.....RETURN

5. YOU MUST TYPE ----- TO FIND OUT WHAT PROGRAMS ARE STORED ON A DISK.

ANSWER QUESTION #5.....CATALOG

QUIZ 2

1. A ----- IS A SERIES OF INSTRUCTIONS
TELLING THE COMPUTER WHAT TO DO.

ANSWER QUESTION #1.....PROGRAM

2. EACH LINE OF A PROGRAM IS ----- SINCE
THE COMPUTER FOLLOWS INSTRUCTIONS IN
NUMERICAL ORDER.

ANSWER QUESTION #2.....NUMBERED

3. THE WORDS OR NUMBERS THAT YOU WANT TO
PRINT MUST BE IN QUOTES AND PLACED AFTER
THE WORD -----.

ANSWER QUESTION #3.....PRINT

4. IT IS PROPER TO END A PROGRAM WITH AN
--- STATEMENT.

ANSWER QUESTION #4.....END

5. WHAT YOU WANT TO PRINT MUST BE IN -----
IN A PRINT STATEMENT.

ANSWER QUESTION #5.....QUOTES

QUIZ 3

1. THE SYMBOL FOR MULTIPLICATION ON THE
COMPUTER IS - . . .

ANSWER QUESTION #1.....*

2. FOUR RAISED TO THE FIFTH POWER MEANS TO
MULTIPLY FOUR BY ITSELF ---- TIMES.

ANSWER QUESTION #2.....FIVE

3. THE SYMBOL FOR DIVISION ON THE COMPUTER
IS - . .

ANSWER QUESTION #3...../

4. 10 PRINT 8*4
THE COMPUTER WILL PRINT THE NUMBER --
AS OUTPUT FOR LINE 10.

ANSWER QUESTION #4.....32

5. T OR F. 10 PRINT "10*3"
THE COMPUTER WOULD PRINT THE NUMBER 30
AS OUTPUT FOR LINE 10.

ANSWER QUESTION #5.....F

QUIZ 4

1. A LETTER THAT STANDS FOR A NUMBER IS CALLED A -----.

ANSWER QUESTION #1.....VARIABLE

2. THE INSTRUCTION TO THE COMPUTER THAT ASSIGNS A NUMBER TO A VARIABLE IS A --- STATEMENT.

ANSWER QUESTION #2.....LET

3. WHEN MORE THAN ONE VARIABLE IS PRINTED A COMMA OR A ----- IS USED TO SEPERATE THE VARIABLES.

ANSWER QUESTION #3.....SEMICOLON

4. A ----- BETWEEN VARIABLES WILL PRINT THE RESULTS ABOUT 15 SPACES APART.

ANSWER QUESTION #4.....COMMA

5. THE COMPUTER WILL PRINT 22 FOR THE LINE
20 PRINT "10 + 12"
IF YOU REMOVE THE -----.

ANSWER QUESTION #5.....QUOTES

QUIZ 5

1. IN THE EXPRESSION $5+4*3^2$
THE COMPUTER WILL DO WHICH FIRST
- 1 $4*3$
 - 2 3^2
 - 3 $5+4$

ANSWER QUESTION #1.....2

2. THE VALUE OF $20-4*2$ IS ANSWER QUESTION #2.....12
3. YOU MUST USE ----- IF YOU WANT THE
VALUE OF $30+10/5$ TO BE 8.

ANSWER QUESTION #3.....PARENTHESES

4. THE VALUE OF $30/2-2^3$ IS -.

ANSWER QUESTION #4.....7

5. THE VALUE OF $3*(7-4)/(2+1)$ IS -.

ANSWER QUESTION #5.....3

QUIZ 6

1. TO ALLOW A PROGRAM TO ASK FOR A NUMBER
AN ----- STATEMENT IS USED.

ANSWER QUESTION #1.....INPUT

2. A ----- MUST FOLLOW THE WORD INPUT.

ANSWER QUESTION #2.....VARIABLE

3. T OR F. IT IS POSSIBLE TO REQUEST MORE
THAN ONE NUMBER AT A TIME WITH ONE
INPUT STATEMENT.

ANSWER QUESTION #3.....T

4. WHAT SYMBOL WILL APPEAR ON THE SCREEN
WHEN THE OPERATOR IS TO INPUT A NUMBER
WHILE RUNNING A PROGRAM WITH AN INPUT
STATEMENT?-----

ANSWER QUESTION #4.....QUESTION MARK

5. TYPE THE WORD ---- TO SEE THE ENTIRE
PROGRAM ON THE SCREEN.

ANSWER QUESTION #5.....LIST

QUIZ 7

1. WHEN YOU WANT TO INPUT A WORD INTO A PROGRAM YOU MUST USE AN ----- VARIABLE.

ANSWER QUESTION #1.....ALPHANUMERIC

2. THE LINE
10 LET A\$=PAUL
IS INCORRECT UNLESS PAUL IS A QUESTION #2.....QUOTES

3. WHICH OF THE FOLLOWING IS NOT AN ALPHANUMERIC VARIABLE?

- 1 A\$
- 2 B
- 3 X\$

ANSWER QUESTION #3.....2

4. WHICH LINE IS CORRECT?

- 1 10 LET G=GUS
- 2 20 LET E\$=EDGAR
- 3 30 LET F\$="FRAN"

ANSWER QUESTION #4.....3

5. WHICH LINE IS CORRECT?

- 1 10 LET C="10"
- 2 20 LET E\$="10"
- 3 30 LET F\$=10

ANSWER QUESTION #5.....2

QUIZ 8

1. TO INSERT REMARKS INTO A PROGRAM A ---
STATEMENT IS USED.

ANSWER QUESTION #1.....REM

2. THE REM STATEMENT WILL HAVE NO EFFECT ON
A PROGRAM BUT IT WILL APPEAR WHEN YOU
---- THE PROGRAM.

ANSWER QUESTION #2.....LIST

3. IN THE LINE
10 PRINT TAB(25);"@"
THE @ WILL BE PRINTED ON THE -- TH SPACE
FROM THE LEFT.

ANSWER QUESTION #3.....25

4. TAB IS ALWAYS USED IN A ----- STATEMENT.

ANSWER QUESTION #4.....PRINT

5. Y OR N. CAN YOU HAVE MORE THAN ONE TAB
IN A PRINT STATEMENT?

ANSWER QUESTION #5.....Y

QUIZ 9

1. A ---- STATEMENT IS USED TO HAVE THE
COMPUTER GO DIRECTLY TO A SPECIFIC LINE.

ANSWER QUESTION #1.....GOTO

2. THE IF MUST BE FOLLOWED BY A -----
IN AN IF...THEN STATEMENT.

ANSWER QUESTION #2.....CONDITION

3. THE THEN MUST BE FOLLOWED BY A
----- IN AN IF...THEN STATEMENT.

ANSWER QUESTION #3.....LINE NUMBER

4. THE COMPUTER SYMBOL FOR NOT EQUAL TO IS
--.

ANSWER QUESTION #4.....(<)

5. IN THE LINES
80 IF Q<40 THEN 10
90 PRINT Q
WHEN Q=30 THE COMPUTER WILL GO FROM
LINE 80 TO LINE --.

ANSWER QUESTION #5.....10

QUIZ 10

1. WHEN A COMPUTER DOES THE SAME THING
OVER AND OVER IT IS CALLED A ----.

ANSWER QUESTION #1.....LOOP

2. YOU CAN MAKE A LOOP WITH A GOTO STATE-
MENT OR A --- NEXT STATEMENT.

ANSWER QUESTION #2.....FOR

3. IN A FOR-NEXT STATEMENT THE FOR IS
ALWAYS FOLLOWED BY A -----.

ANSWER QUESTION #3.....VARIABLE

4. YOU MUST USE THE WORD ---- TO INCREMENT
THE VARIABLE BY 4 IN A FOR-NEXT
STATEMENT.

ANSWER QUESTION #4.....STEP

5. T OR F. THE VARIABLE IN A FOR-NEXT
STATEMENT MUST ALWAYS START AT 1.

ANSWER QUESTION #5.....F

QUIZ 11

1. THIS LET STATEMENT IS AN EXAMPLE OF A
----- STATEMENT.
20 LET J=J+1

ANSWER QUESTION #1.....COUNTER

2. THIS LET STATEMENT IS AN EXAMPLE OF AN
----- STATEMENT.
30 LET R=R+M

ANSWER QUESTION #2.....ADDER

3. THE VARIABLE IS ALWAYS INCREASED BY A
----- NUMBER EACH TIME A LOOP IS
PERFORMED IN A COUNTER STATEMENT.

ANSWER QUESTION #3.....CONSTANT

4. FOR THE ADDER STATEMENT
40 LET F=--+ A
THE BLANK MUST BE THE VARIABLE --.

ANSWER QUESTION #4.....F

5. COUNTER AND ADDER STATEMENTS ARE ALWAYS
FOUND IN --- STATEMENTS.

ANSWER QUESTION #5.....LET

FINAL QUIZ

1. THE STATEMENT THAT HELPS THE PROGRAMMER
REMEMBER WHAT A PROGRAM DOES IS ---.
ANSWER QUESTION #1.....REM
2. THE SYMBOL FOR MULTIPLICATION ON THE
COMPUTER IS --.
ANSWER QUESTION #2.....*
3. THE WORD USED TO STOP THE COMPUTER UNTIL
THE OPERATOR PUTS IN A NUMBER OR A WORD
IS -----.
ANSWER QUESTION #3.....INPUT
4. THE WORD USED TO CLEAR THE MEMORY ON THE
COMPUTER IS ---
ANSWER QUESTION #4.....NEW
5. 10 PRINT HELLO
FOR LINE 10 TO PRINT THE WORD HELLO THE
HELLO MUST BE IN -----.
ANSWER QUESTION #5.....QUOTES
6. A PART OF A PROGRAM THAT IS DONE OVER
AND OVER AGAIN IS CALLED A ----.
ANSWER QUESTION #6.....LOOP
7. THE EXPRESSION $8+5*3^2$ EQUALS --.
ANSWER QUESTION #7.....51
8. 40 LET B=B+X
LINE 40 IS AN EXAMPLE OF AN -----
STATEMENT.
ANSWER QUESTION #8.....ADDER
9. THE VARIABLE B\$ IS CALLED AN
----- VARIABLE.
ANSWER QUESTION #9.....ALPHANUMERIC
10. 60 IF B=7 ---- 60
THE WORD LEFT OUT IN LINE 60 IS ----.
ANSWER QUESTION #10.....THEN
11. A ----- BETWEEN VARIABLES WILL SPACE
THEM ABOUT 15 SPACES APART WHEN PRINTED.
ANSWER QUESTION #11.....COMMA
12. 10 ----- "RANDY LOVES MATH."
THE MISSING WORD IN LINE 10 THAT WILL
PRINT THE SENTENCE IN QUOTES IS -----.
ANSWER QUESTION #12.....PRINT

13. 10 --- X=7
20 --- B=B+1
THE MISSING WORD IN LINES 10 AND 20
IS ---

ANSWER QUESTION #13.....LET

14. YOU MUST TYPE --- TO EXECUTE THE PROGRAM
IN THE COMPUTER'S MEMORY.

ANSWER QUESTION #14.....RUN

15. TO DELETE A LINE FROM A PROGRAM YOU MUST
TYPE THE ---- OF THE LINE.

ANSWER QUESTION #15.....LINE NUMBER

16. THE SYMBOL FOR GREATER THAN OR EQUAL TO
ON A COMPUTER IS --.

ANSWER QUESTION #16.....>=

17. 10 PRINT --- (18) "%"
THE MISSING WORD THAT WILL PRINT THE %
ON THE 18TH SPACE FROM THE LEFT IS ---.

ANSWER QUESTION #17.....TAB

18. THE WORD USED TO CLEAR THE SCREEN ON THE
APPLE IS ----.

ANSWER QUESTION #18.....HOME

19. THE WORD THAT IS USED TO CONCLUDE MOST
PROGRAMS IS ---.

ANSWER QUESTION #19.....END

20. THE COMPUTER SYMBOL FOR AN ----- IS ^
OR

ANSWER QUESTION #20.....EXPONENT

21. AFTER EVERY COMMAND YOU MUST HIT THE
ENTER OR ----- KEY.

ANSWER QUESTION #21.....RETURN

22. 10 FOR N=6 TO 99 STEP 3
20 PRINT N
30 NEXT N
FOR LINES 10-30 THE FIRST VALUE OF N TO
BE PRINTED IS 6. THE SECOND VALUE OF N
TO BE PRINTED IS --.

ANSWER QUESTION #22.....9

23. TYPE THE WORD ---- TO SEE ALL THE LINE
NUMBERS AND STATEMENTS FOR THE PROGRAM
IN THE COMPUTER'S MEMORY.

ANSWER QUESTION #23.....LIST

24. 10 FOR A=1 TO 7
20 PRINT A
30 ---- A
THE MISSING WORD IN LINE 30 IS ----.

ANSWER QUESTION #24.....NEXT

25. PROGRAMS MAY BE SAVED ON A CASSETTE TAPE
OR ON A ----.
ANSWER QUESTION #25.....DISK
26. TYPE ----- TO SEE WHAT PROGRAMS ARE
SAVED ON THE DISK THAT YOU ARE USING.
ANSWER QUESTION #26.....CATALOG
27. A ---- STATEMENT WILL SEND THE COMPUTER
TO A SPECIFIC LINE.
ANSWER QUESTION #27.....GOTO
28. A LETTER THAT STANDS FOR A NUMBER IS
CALLED A -----.
ANSWER QUESTION #28.....VARIABLE
29. THE FIRST WORD OF A STATEMENT USED TO
FORCE THE COMPUTER TO MAKE A DECISION
IS --.
ANSWER QUESTION #29.....IF
30. T OR F. A FLAG IS A COMMAND USED TO
STOP A PROGRAM.
ANSWER QUESTION #30.....F

ENRICHMENT EXERCISES

The following problems were selected as extra programming aids. Some are easy. Most are challenging or very difficult. Students should be encouraged to look over these problems and select the one or ones that interest them. Answers are not provided. It is hoped that the students and teacher can come up with creative solutions. Don't be afraid to try these!

1. Toss a die several times. Enter the number of ones, the number of twos, the number of threes, and so on, that turn up. Print the percent of tosses that are ones, the percent that are twos, and so on.

2. Given four real numbers, print them from largest to smallest. Do not assume that the numbers are listed in the data in any special order.

3. Input a real number X. Print X, its additive inverse, and (if it has one) its multiplicative inverse.

4. Given the number of a baseball team's wins and losses, compute its winning percentage. Assume that there were no ties.

5. Input the number of a team's wins, losses, and ties, and print its winning percentage. Assume that a tie game counts as a full game played and a half-game won.

6. Students in a certain course take two examinations. If a score is less than 40 on either exam, then the student fails the course. If the average on the two exams is below 60, the student also fails the course. An average of 60 to 80, inclusive, is passing (without honors). An average above 80 is passing with honors. Given a student's scores on the two exams, print whether he or she fails, passes (without honors), or passes with honors.

7. Given the lengths of any two sides of a right triangle, the Pythagorean Theorem enables you to compute the third side. Write a program which finds the length of the missing side, whether it is a leg or the hypotenuse. One way to accomplish this is to INPUT A,B,C but enter 0 for the unknown side. Then test whether A=0, B=0, or C=0 and branch to one of three assignment steps to compute the missing length and print it.

8. Input the radius of a sphere and compute its volume. Formula:
 $V = 4/3 * \pi * R^3$

9. Input L, the length of one side of an equilateral triangle, and compute the perimeter and area.

10. Input the measure of the vertex angle of an isosceles triangle. Print the measure of a base angle of the triangle.

11. Input three positive numbers A, B, and C. Determine whether these could be the lengths of the sides of a right triangle. (Note: You must first check that A, B, and C can be sides of ANY triangle.)

12. Input N, a positive integer denoting the number of sides of a polygon. Assume that the N-gon is regular and print the measure of each interior angle. Formula: $M = (N-2) * 180 / N$

13. Given the coordinates of three points in the XY-plane, determine whether the points are collinear.

14. Compute the perimeter of a right triangle, given the lengths of the two legs.

15. Compute the perimeter of a right triangle, given the length of the hypotenuse and the length of one leg.

16. Assume that the cost, not including tax, for sending a telegram from Chicago to Los Angeles is \$4.50 for the first fifteen words or less, plus \$0.15 for each additional word beyond fifteen. Write a program that, given the number of words in a telegram (Chicago-Los Angeles), computes the cost of the telegram.

17. Assume that a baby-sitter works for a rate of \$1.20 per hour until 11 p.m. and \$1.80 per hour thereafter. Given the time that he or she begins and the time that he or she ends, compute the fee for an evening's work. (Hint: Input the times in terms of the twenty-four hour clock; e.g., for 7 p.m. input 1900.)

18. Input the sales and expenses of the Acme Trucking Company for the past year. Calculate the profit as sales minus expenses. Then compute the tax by the following rule:
If $0 < \text{profit} \leq \$20,000$, the tax is 30% of the profit;
If $\text{profit} > \$20,000$, the tax is 30% of the \$20,000 plus 50% of the profit above \$20,000;
If $\text{profit} \leq 0$, the tax is zero.
Print the profit, the net tax, and the ratio of the tax to the profit.

19. Liberty Bank imposes a service charge based on the amount in an account at the end of the month.

Balance at end of month.....	Service charge
\$0-\$99.....	\$.75
\$100-\$199.....	\$.50
\$200-\$299.....	\$.25
\$300 or more.....	none

Given the ending balance in an account and the account number, print the service charge and the net amount in that account.

20. Given the population of a city, state, or country at the beginning of a year and the population at the end of the year, print the percent of change in population. If there is a gain, print the message XX% INCREASE. If there is a loss, print XX% DECREASE.

21. Given a baseball player's times at bat and number of hits, write a program to compute the player's batting average.
22. The 'slugging percentage' of a baseball hitter is the ratio of his total bases to his at-bats. Given his number of singles, doubles, triples, home runs, and at-bats, print a player's slugging percentage. Note: Like batting average, slugging percentage is expressed as a three-place decimal. A slugging percentage over .500 is considered outstanding.
23. A coin is flipped repeatedly. Input the number of heads and the number of tails that occur. Compute the percent of the total flips that are heads and the percent that are tails.
24. Given a linear equation $AX+B=C$ ($A \neq 0$), input A, B, and C and solve for X. For example, if the equation is $7X+3=19$, print $X=2$.
25. A baseball pitcher's "earned-run average" is computed using the formula $ERA=(R*9)/I$ where R=number of earned runs allowed and I=innings pitched. Given R and I, compute and print a pitcher's ERA. Assume that a whole number of innings is pitched.
26. Compute a baseball player's "fielding percentage", given his put-outs, assists, and errors.
fielding percentage=chances handled successfully / total chances = $(P+A)/(P+A+E)$
27. Given the list and net prices of an item, find the amount of discount and the percent (rate) of discount.
28. Given the cost price and the percent of margin (markup) of an item, calculate the selling price and the gross profit.
29. Calculate a person's gross weekly pay, given the rate per hour, overtime rate per hour, number of regular hours worked that week, and number of overtime hours that week.
30. Input the cash selling price of an item, the down payment, the monthly payment, and the number of monthly payments to be made. Find the finance charge.
31. Given the unpaid cash balance of an item, the finance charge, and the number of months of payments, calculate the monthly payment.
32. Compute the annual depreciation rate of an investment. Given is the original value of the investment, its current value, and its age in years. Use the formula:
depreciation rate = $1-((\text{current value}/\text{original value})^{(1/\text{age})})$
33. Assume that one ounce of hamburger contains 80 calories, that a bun has 200 calories, and that one French fry has 14 calories. Given the number of ounces of hamburger, whether a person has a bun (1=yes, 0=no), and the number of French fries, calculate the number of

calories in a person's meal.

34. Given the height from which an object falls, determine the time it will take to fall. use formula:
 $\text{distance}(\text{feet}) = 16 * (\text{time}^2)$

35. Use a variety of values for A and B to test the truth of the following:

$$|A + B| = |A| + |B|$$

$$|A * B| = |A| * |B|$$

$$|A - B| = |B - A|$$

36. In a certain city if you get a speeding ticket, your fine is based on the amount by which you exceeded the speed limit.
 Amount over limit(miles per hour).....Fine

1-10 mph.....	\$5
11-20 mph.....	\$10
21-30 mph.....	\$20
31-40 mph.....	\$40
above 41 mph.....	\$80

Given the speed limit and the speed arrested at, print the fine.

37. Suppose that discounts for direct-dial long-distance calls are figured as follows (where the times are at the calling points).

Time.....	Day.....	Discount
7am-5pm.....	weekdays.....	none
5-11pm.....	any day.....	20%
11pm-7am.....	any day.....	40%
7am-11pm.....	Saturday.....	20%
7am-5pm.....	Sunday.....	40%

Suppose that the charge for a direct-dial call between New York and Los Angeles is \$.84 per minute. (61 seconds count as a two-minute call.) Use the discount rates listed above. Given the day of the week and the time at which the call originated and the number of seconds the call lasted, print the charge.

38. Density (d) is mass (m) per unit volume (v). Given values for any two of the variables, find the third. Input values for all three variables but with zero for the unknown one. Formula: $d=m/v$

39. Given an integer A and a positive integer N, compute A^N without using exponentiation.

40. Find every two-digit whole number that equals the sum of the squares of its digits.

41. Find the sum of the squares of any ten consecutive integers.

42. Find the LCM (least common multiple) of two integers.

43. Find the LCM of three integers.

44. Find the GCF (greatest common factor) of two integers.

43. Find the GCF of three integers.

46. Suppose that you purchase the Newtown Spears, a minor league baseball team, for \$72,000. The league tells you that the profit (in \$1000 units) of the franchise can be projected for the next eight years by using the formula:

$$P = (T^3) - (5 \cdot T^2) + (10 \cdot T) - C$$

where P=profit, T=time in years, C=cost in thousands (72). Calculate and print your profit or loss for each of the next eight years. Also calculate and print your net profit (or loss) at the end of eight years.

47. Show the transaction during one month for a person's bank account. (In effect you are writing a checkbook-balancing program.) Enter the balance at the beginning of the month. Then input the transaction in chronological order, entering first the day number of the transaction, a code to indicate whether it was a debit or credit, and the amount. Use a code number to signal the last transaction of the month. Have the computer print a record of the month in four columns as follows.

DAY.....DEBIT.....CREDIT.....CURRENT BALANCE

At the bottom print the balance at the end of the month.

48. Expand exercise 47 to include a bank service charge. The charge is \$.25 per month but at any time during the month the balance falls below \$300, an additional \$.25 is debited. (Thus the maximum monthly service charge is \$.50).

49. Given the net profit and the number of partners of a company, enter the percent of ownership of each partner. Print the profit of each partner.

50. Write a program that will give the students in your class a quiz on the information covered in your current unit in Social Studies. Have at least 5 questions. Be sure your program prints the student's score when he or she is finished.

51. Elizabeth is a real estate agent. She sold a house for \$47,500. Her commission is 7% of the selling price. Write a program that will compute the commission for this, or any other, sale.

52. Write a program to find the circumference of any circle. Use the formula $C = 2 \cdot \pi \cdot R$ where $\pi = 3.14$ and $R = \text{radius}$.

53. Write a program that will compute statistics for your school basketball team. Print the results out in a table with the following headings: NAME, TOTAL POINTS SCORED, FIELD GOAL PERCENTAGE, FREE THROW PERCENTAGE. Calculate field goal percentage and free throw percentage to the nearest thousandth. Other programs to compute statistics for football, soccer, volleyball, etc. may be written.

54. Jeff ran last week's marathon in 222 minutes. Find his time in minutes per mile. A marathon is 26.2 miles.

55. Write a program that will calculate average miles per hour given starting time, finishing time, and miles traveled. (Note: Use the twenty-four hour clock to compute time. e.g., 8 p.m. is 2000 hours.)

56. You drove from Syracuse to Buffalo on the New York State Thruway. The toll charge was \$3.20. The number of miles traveled was 220. Write a program to find the cost per mile.

57. Suppose you were to fold a piece of paper a whole bunch of times. Each successive fold should produce a piece of paper twice as thick as the previous one. Write a program to figure how thick the paper will be after N folds. Assume the unfolded thickness of the paper to be .01 inches. Be sure to convert inches to feet when you exceed twelve and then feet to miles when you exceed 5280 feet. The output should include, in a table, the number of folds and the thickness. A geometric series whose general term is $2^N \Delta N$ might prove helpful.

58. Suppose a man came to work for you. He didn't want to be paid like everyone else. He asked for a special system of payment based on doubling. He wanted to be paid twice a month for a total of 24 pay periods per year. He wanted only \$.01 for his first pay period, \$.02 for his second pay period, \$.04 for his third, and so on, each time doubling his previous pay check. Program your computer to print out the man's salary for each pay period along with a running total of how much he has received to date.

59. Write a program to help the Indians out. The Indians in question are the ones who sold Manhattan Island to the Dutch for the paltry sum of \$24.00. The sale took place in the year 1626. Suppose they had deposited the money in the local bank. Interest rates changed according to the table given below according to century.

1600's.....	2%
1700's.....	3%
1800's.....	4%
1900's.....	5%

Figure out how much the Indians would have in the bank today. Compound the interest annually.

60. Can you find a five digit number which when multiplied by four has its digits reversed? What you want is a number ABCDE such that:

$$4 * ABCDE = EDCBA$$

A knowledge of place value and expanded notation may prove helpful here.

INSTRUCTIONS FOR COMPUTER QUIZ MAKER PROGRAM by DOUGLAS W. GREEN - CORTLAND SCHOOLS

The purpose of this program is to allow an inexperienced programmer to create quizzes that will run on a microcomputer. The questions and the correct answers are typed into data statements at the end of the program where they are read one at a time and presented to the student. If the student enters an incorrect response, the program remembers it and after all questions have been answered, the questions that were missed are repeated. Questions continually missed are repeated until they are answered correctly. After a question is answered incorrectly the computer prints the correct answer at the top of the screen. After every answer is input the current percentage correct is also printed. Correct answers are greeted by a positive reinforcement message that is selected at random from six available messages.

INSERTING QUESTIONS: Questions and answers created by the user are placed into data statements at the end of the program. The first question and answer can be placed in statement number 1010 and subsequent questions and answers can be placed in data statements having higher statement numbers. The last data statement should look like this:

2001 DATA DONE,DONE. This is a flag that tells the computer that the last question has been read. You can put more than one question and answer in a single data statement if you wish but remember that long statements are more difficult to edit. The correct syntax for a sample statement would look like this:

statement number DATA question,answer

Since a comma is used to separate the question from the answer, you cannot use a comma in either your question or your answer. This means that your questions must be composed of short simple sentences rather than compound questions to place on each quiz, consider the level of the students you are dealing with. You should also watch a few groups as they take the quizzes to see if the length and difficulty are appropriate.

SOME TIPS ON WRITING QUESTIONS: Since the computer will only recognize the exact answer, you should use questions that have very definite answers. The best type for this purpose are multiple choice questions. Unfortunately they are also the most difficult to write. Other question types that work well are those requiring yes-no or true-false replies. With this type of question you should tell the student what type of responses are expected. A message in parentheses can save some confusion on the part of the student. Here is an example.

1010 DATA A STATEMENT PROVIDING OUTPUT CONTAINS THE WORD PRINT (T OR F),T

You can use fill in the blank questions as long as there is only one correct answer. In order to give the student a little help you can use the - symbol to indicate the number of letters in the missing

word like so:

1020 DATA STATEMENTS USED TO HELP THE PROGRAM THAT ARE IGNORED BY THE COMPUTER CONTAIN THE WORD ---,REM It is possible to avoid having portions of two words printed on two lines on your screen. This does require that you pay attention to the length of the question as you are entering it. Imagine a vertical line at the left edge of the first letter in the first word of your question. Do not allow any subsequent words to start on one side of this line and end on the other side. Such words will be broken in two when printed. A single question can contain about 235 characters. If you begin to hear a beeping noise stop typing. The maximum number of characters in a single statement is 255. The beep tells you that you are approaching this limit. If you go over 255 characters the entire statement will be lost from the computer's memory. If this happens you can save yourself some typing by entering the escape mode and tracing over the portion of your statement that you wish to save.

SPECIFIC DIRECTIONS: 1) Load the quiz model program 2) Add your questions and answers as data statements according to the directions given above 3) Run this program to make sure it works. Try to answer all of the questions correctly. If you are greeted with an incorrect response even though it looks identical to your answer from the data statement just retype the data statement in question. It is possible to enter control characters without knowing it. Such characters will cause problems when the program is run. 4) Save your program using a name that describes the quiz. 5) To make another quiz clear the memory and goto direction #1.

DIRECTIONS FOR QUIZ MODEL MULTIPLE CHOICE

This program allows you to write multiple choice quizzes. For each question, it selects three answers from other questions at random and presents them as wrong answers. The three wrong answers and the correct answer are then scrambled and presented as the four choices. The program requires that the user press one of the keys from 1 through 4. If any other key is pressed the program will assume that a wrong answer has been given. The return key need not be pressed. If it is pressed it will also result in the program assuming that a wrong answer has been entered.

After all questions have been answered one time, the final percentage is printed and the questions that were missed are presented again. Questions missed a second time are brought back again and again until they are answered correctly. If a question is answered incorrectly, the correct answer is printed on the screen along with the question until the user hits the space bar.

The fact that wrong answers are selected at random and that the choices are randomly mixed means that the quizzes look different each time through. Repeated questions will also contain different wrong answers each time they are repeated.

Since all answers are used as distractors, the answers to all questions should be similar enough so that they make convincing distractors. Some areas that work well are math questions and vocabulary questions in all languages. The teachers must check to be sure that no two questions have the same answer. Any group of questions that have single word answers all tend to work well. Questions that require either true-false or yes-no responses, should be avoided. If the questions you wish to use do not fit this model, use the other quiz program included with this package. You should also study the directions for entering your questions and answers as data statements at the end of the program. Also be sure to include as your final data statement one that looks like this:

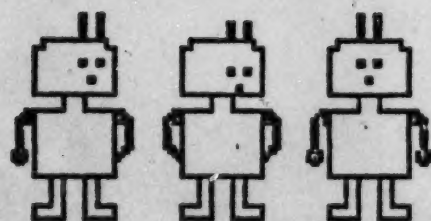
3000 DATA DONE,DONE.

This provides the program with the flags that it needs to know when it is finished. Finally, be sure to enter at least four questions before you try to run the program. I suggest a 10 question minimum to make this program useful. To see how it works you can run the programs called: COMPUTER QUIZ 1, COMPUTER QUIZ 2, COMPUTER QUIZ 3, and BERLITZ FRENCH.

SPECIFIC DIRECTIONS:

- 1) LOAD QUIZ MODEL - MULTIPLE CHOICE 2) 105 return (erases statement #105)
- 3) Add your questions and answers as data statements at the end of the program. LIST to find the end.
- 4) RUN the program to see how it works.
- 5) Correct the questions that need it.
- 6) SAVE the program by a name that describes the quiz. Be sure to save it on the disk where you want it.
- 7) To start another quiz DEL the statements containing your questions and answers and start again. Be sure to save the second quiz by a different name.

B
U
Z
Z
Z
Z
Z
Z
Z
Z
Z
Z
Z
Z
Z
Z
W
O
R
D
S



BUZZZZZZZZZZZZZZWORDS

ACOUSTIC COUPLER: A device that uses telephone lines. You need one at each end. Don't overlook this cost.

ALPHANUMERIC: A variable that may include letters of the alphabet, numerals, and other symbols. Also called a string variable.

ASSEMBLER: A program that translates assembly language into machine language.

ASSEMBLY-LANGUAGE: A programming language composed of three letter codes. It is more difficult than high level languages like BASIC but is easily assembled into machine language for faster program execution.

BASIC: Beginners All-purpose Symbolic Instruction Code. The high level language used in most beginning computer programming courses.

BINARY NUMBER SYSTEM: A number system that uses the numbers 0 and 1 to represent all numbers.

BIT: The amount of memory space capable of storing one piece of binary data like a 1 or a 0. Similar to an on-off switch.

BOOT: A process that prepares a computer to communicate with a disk drive. A system must be "booted" before programs can be saved.

BUG: An error in a program.

BYTE: The amount of memory space capable of storing one character of information. Usually composed of 8 bits.

CAI: Computer Aided Instruction. A method of instruction where the student has the opportunity to interact with a computer.

CARD READER: A device that reads programs and/or data from punched cards.

CHARACTER: A number, letter, or symbol on the computer keyboard.

CMI: Computer Managed Instruction. One of many methods for using a computer for storing and processing student performance data.

COMPILER: A device that translates the program from a high level language into machine language. It can be stored on disk or tape. Assemblers and interpreters do this same sort of thing.

COBOL: Common Business Oriented Language. A high level language used in most business applications. Easily understood by

non-programmers, but more difficult to learn than BASIC.

CORE: The amount of memory space located in the computer itself. It is measured in bytes. 8k worth of core can store 8000 bytes. A misnomer for today's systems. They do not contain magnetic cores to store data.

CPU: Central Processing Unit. The location in the computer where the program directions are executed.

CRT: Cathode Ray Tube. Better known as a TV screen. This type of device saves paper, but you can not take the output home with you.

CURSOR: An indicator that shows the location of the next character to be printed.

DEBUG: The process of correcting errors in a program.

DISK DRIVE: A unit that can store and retrieve information from a magnetic disk. More expensive than a tape drive, but much faster.

DISKETTE: A 5 1/4 inch diameter magnetic disk used to store data. VERY FRAGILE...HANDLE WITH CARE!!!

FILE: collection of related records treated as a unit. A set of information on your employees could represent your payroll file.

FLAG: An indicator used to tell some later part of a program that some condition occurred earlier.

FLOWCHART: A diagram used to illustrate the program logic.

FORTRAN: FORMula TRANslator. A high level language used primarily in science and engineering. It does complex calculations rapidly.

HARDCOPY: A printed copy of computer output. A must for business or programming use. Printer cost may not be included unless asked for.

HARDWARE: The mechanical and electronic pieces of equipment that comprise the computer system.

HEXADECIMAL SYSTEM: A system using 16 digits (numbers 0 - 9 and letters a - f) instead of 10 for counting purposes. This system is used extensively in computer programming in assembly language.

HIGH LEVEL LANGUAGE: Any one of a number of programming languages that rely on English language words. COBOL, BASIC, Pascal, PL/1, APL, and FORTRAN for example.

HIGH RESOLUTION GRAPHICS: Ability to draw anything on a grid of 280 horizontal positions by 192 vertical positions. These values may change with various computers.

INPUT: Entering data into the computer's memory.

INTEGRATED CIRCUIT: Also known as IC's or chips. These components perform the basic tasks for the computer. ROM, RAM, and CPUs are IC's. They contain thousands of transistors on a single chip.

INTERFACE: A connection between two pieces of hardware. May look like fancy cable. Can be a problem when two companies provide different parts of your system.

I/O DEVICE: Input or output device. Examples include: printer, CRT, tape or disk drive units, card or punched tape reader, keyboard, or plotter.

MACHINE LANGUAGE: A set of symbols and rules for combining them that conveys instructions to the computer. Uses no English words and needs no compiler.

LOW RESOLUTIONS GRAPHICS: The ability to draw any thing on a grid 40 columns across by 48 rows down. These values may change on various computers.

MAIN FRAME: The main part of a larger computer. Often a large computer that controls many terminals.

MICRO: Short for micro computer. These recent inventions are typewriter-size self contained computers that are being used in schools, small businesses, and homes. The major brands include: RADIO SHACK (TRS-80), APPLE, PET, ATARI, NORTH STAR, and OHIO SCIENTIFIC.

MODEMS: Performs the same task as an acoustic coupler without the use of the telephone.

NUMERIC: A variable containing numbers only.

OUTPUT: The results of a computer's processing. Output can be sent to a CRT or a printer to be read or to a disk to be stored.

PASCAL: High level language similar to BASIC. It requires a more structured approach to programming and as a result runs faster.

PERIPHERAL: An input or output device that is not part of the main computer console. Find out what peripherals can be added before you buy.

PROGRAM: A step by step set of directions that the computer follows while doing its job. Many businesses buy the programs they need. So do many schools.

RAM: Random Access Memory. Memory that can be accessed at any point. A common property of computer core memory and disk-drive

memory.

ROM: Read Only Memory. Memory that must be read from start to finish. A drawback of information stored on tape or computer cards. ROM in the CPU is not erased when the computer is turned off.

SOFTWARE: The program that provides instructions for the computer.

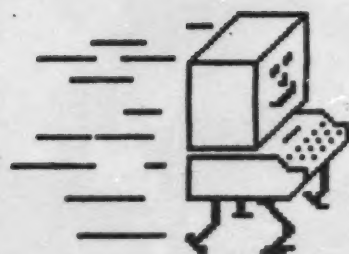
STRING: A symbol or group of symbols that can be printed as is or set equal to an alphanumeric variable.

SUBROUTINE: A smaller program used within a program. When the same task must be performed in more than one location.

TAPE-DRIVE: A unit that can store and retrieve programs and data from magnetic tape. Each data file is read in the same order each time.

TERMINAL: A point where information can enter or leave the computer. It can contain CRT or a printer. Terminals can be placed in the same building as the computer. They can also be connected by phone lines to computers that are far away.

VARIABLE: A symbol that stands for a number. Since the value of the number can vary, the symbol is called a variable.



Diskette Care & Handling

To get the best performance from your diskettes, and to provide the best protection to the information you put on them, we recommend that you learn, remember and always live by these Diskette Do's & Don'ts:

A. Whenever handling your diskettes, DON'T ever touch the diskette recording surface. Handle your diskettes only on the jacket area.

B. After you finish using a diskette, DO put it back in its protective envelope immediately. If you leave a diskette exposed: dust, debris, cigarette smoke and other environmental hazards will quickly cause damage to your data.

C. When writing on your diskette ID label, DON'T use a ball point pen, fountain pen, pencil or other hard marker. Use only a felt tip pen. Hard markers can make an impression on your diskette, causing data loss.

D. Diskettes work best at temperatures between 50° F and 122° F. Temperatures above 122° F can occur easily in an automobile, or in front of a window pane. DON'T store your diskettes in areas of extreme temperature. To keep them working best, keep them where the temperature's comfortable.

E. DON'T let a magnet or magnetized object get near your diskettes. That little magnet that keeps all your phone messages in place could cause your diskette to lose its information.

F. DON'T ever fold, bend or crease your diskettes. Handle your diskette carefully, so it will always be in good shape to make contact with the read/write head properly.

G. DON'T use erasers on the diskette ID

label. You could get debris in the diskette, and that's a lot worse than having a messy ID label.

H. Do gently load your diskette in your drive unit. If you don't it may bend or center improperly. That causes the diskette to rotate in an elliptic orbit, missing data.

BACK-UP YOUR IMPORTANT INFORMATION

Anytime you find yourself with information on a diskette that you simply can't lose, you're open for trouble. Why? Because you could accidentally lose it. Think of all the time you put into creating a specific program, or worse yet, the money you spent buying one. In a case like this, the value of a back-up diskette can't be over-emphasized. In some instances, it may be advisable to have two back-ups. Make sure you store your back-up data in a well-protected, safe place and use your write protect feature if available.

PROTECT YOUR DATA WITH WRITE PROTECT

Most systems today employ the use of a write protect feature. Simply stated, this means your system cannot accidentally write over or destroy the information contained on your diskette when the "Write Protect" feature is activated.

To "write protect" data on your minidisks, the write protect notch must be covered-up with one of the foil labels supplied. To write on a minidisk, uncover the notch.

